

Reference Material

Oct 13, 2015

Reference Material

Use the reference information in this section to get an in-depth understanding of the following NetScaler components:

NetScaler SNMP OIDs - Details of the SNMP OIDs that can be used to obtain information from a NetScaler appliance.

NetScaler Syslog Messages - Details of the Syslog messages given by the NetScaler appliance.

NetScaler CLI Commands - Details of the commands that can be used to configure the NetScaler appliance through the CLI. You can also view the details of each command in the NetScaler CLI, by entering the "man <ns-command-name>" command.

Policy Expressions - Details of the policy expressions available on the NetScaler.

Quick Start Guides - A reference to quick installation and configuration of your hardware appliance.

NetScaler SNMP OID Reference

A detailed list of the SNMP OIDs that can be used to obtain information from a NetScaler appliance.

- o Generic MIB-II OIDs

- system
- snmp
- interfaces
- ifTable
- ifMIBObjects
- ifXTable
- icmp
- udp

- o NetScaler Enterprise OIDs

- nsSysGroup
 - nsFeatureInfo
 - nsModelInfo
 - nsHighAvailabilityGroup
 - vlanTable
 - nsIpAddrTable
 - nsResourceGroup
 - nsCPUtable
 - nsSysHealthTable
 - nsSysHealthDiskTable
 - nsIpStatsGroup
 - nsIcmpStatsGroup
 - nsUdpStatsGroup
 - nsTcpStatsGroup
 - nsSslStatsGroup
 - nsHttpStatsGroup
 - nsCacheStatsGroup
 - nsCompressionStatsGroup
 - nsIfStatsTable
 - nsExpressionTable
 - htmlInjectionStatsGroup
 - nsSslVpnStatsGroup
 - nsAaaStatsGroup
 - nsGlobalConfigSettings
 - nsInetAddressTable
 - nsNicStatsGroup
 - clusterTable
 - nsClusterStatsGroup
 - nsIp6StatsGroup
 - nsTdlInetAddressTable
 - nsCaStatsGroup
 - nsvPathStatsGroup
 - vxlanTable
 - cacheGroupTable
 - aclStatsGroup
 - nsAclTable
 - saclStatsGroup
 - acl6StatsGroup
 - nsAcl6Table
 - pbrStatsGroup
 - nsPbrTable
 - sacl6StatsGroup
 - pbr6StatsGroup
 - nsPbr6Table
 - gslbGlobalStats
 - nsPolicyStatsTable
 - nsDnsServerStatsGroup
 - nsdnsRegisterTable

- scPolicyStatistics
- sslCertKeyTable
- sslCrITable
- sslCipherGroupTable
- dosPolicyTable
- dosPolicyStatistics
- pqPolicyConfigTable
- pqPolicyStatistics
- crPolicyMapConfigTable
- appFirewallStatistics
- appfwProfileTable
- nsRnatGlobalStats
- nsRnatPerIPStatsTable
- piPolicyTable
- nsInatGlobalStats
- nsInatPerNat46StatsTable
- nsInatPerNatStatsTable
- nsNat64GlobalStats
- nsLLDPConfigGroup
- nsLLDPStatsGroup
 - nsLLDPStatsTxPortTable
 - nsLLDPStatsRxPortTable
- nsLLDPLocSystemsGroup
 - nsLLDPLocPortTable
 - nsLLDPLocManAddrTable
- gslbSitesTable
- gslbPoliciesTable
- nsDomainTable
- scPolicyConfigTable
- nsLLDPRemTable
- nsLLDPRemManAddrTable
- serviceTable
- serverTable
- serviceScpolicyTable
- serviceAdvanceSslConfigTable
- serviceCipherBindingTable
- serviceGlobalStatsGroup
- serviceGroupMemberTable
- serviceDospolicyTable
- monitorMemberTable
- monServiceMemberTable
- serviceGroupTable
- vserverTable
- vserverServiceTable
- vserverCspolicyTable
- vserverCrpolicyTable
- vserverGlobalStatsGroup
- lbvserverTable
- vserverPqpolicyTable
- vserverScpolicyTable
- vserverAdvanceSslConfigTable
- vserverCipherBindingTable
- vserverCsPiPolicyTable
- snmpTrapVarBindOidsGroup
- Generic MIB-II Traps
- NetScaler Enterprise Traps

Note: Refer to the Hardware documentation for the recommended range for the hardware attributes.

Generic MIB-II OIDs

system (1.3.6.1.2.1.1)

sysDescr (1.3.6.1.2.1.1.1)

A textual description of the entity. This value should include the full name and version identification of the system's hardware type, software operating-system, and networking software.

sysObjectID (1.3.6.1.2.1.1.2)

The vendor's authoritative identification of the network management subsystem contained in the entity.

This value is allocated within the SMI enterprises subtree (1.3.6.1.4.1) and provides an easy and unambiguous means for determining 'what kind of box' is being managed. For example, if vendor 'Flintstones, Inc.' was assigned the subtree 1.3.6.1.4.1.424242, it could assign the identifier 1.3.6.1.4.1.424242.1.1 to its 'Fred Router'.

sysUpTime (1.3.6.1.2.1.1.3)

The time (in hundredths of a second) since the network management portion of the system was last re-initialized.

sysContact (1.3.6.1.2.1.1.4)

The textual identification of the contact person for this managed node, together with information on how to contact this person. If no contact information is known, the value is the zero-length string.

sysName (1.3.6.1.2.1.1.5)

An administratively-assigned name for this managed node. By convention, this is the node's fully-qualified domain name. If the name is unknown, the value is the zero-length string.

sysLocation (1.3.6.1.2.1.1.6)

The physical location of this node (e.g., 'telephone closet, 3rd floor'). If the location is unknown, the value is the zero-length string.

sysServices (1.3.6.1.2.1.1.7)

A value which indicates the set of services that this entity may potentially offer. The value is a sum.

This sum initially takes the value zero. Then, for each layer, L, in the range 1 through 7, that this node performs transactions for, $2^{(L-1)}$ is added to the sum. For example, a node which performs only routing functions would have a value of 4 ($2^{(3-1)}$). In contrast, a node which is a host offering application services would have a value of 72 ($2^{(4-1)} + 2^{(7-1)}$). Note that in the context of the Internet suite of protocols, values should be calculated accordingly:

layer functionality

1 physical (e.g., repeaters)

2 datalink/subnetwork (e.g., bridges)

3 internet (e.g., supports the IP)

4 end-to-end (e.g., supports the TCP)

7 applications (e.g., supports the SMTP)

For systems including OSI protocols, layers 5 and 6 may also be counted.

snmp (1.3.6.1.2.1.11)

snmplnPmts (1.3.6.1.2.1.11.1)

The total number of messages delivered to the SNMP entity from the transport service.

snmplnBadVersions (1.3.6.1.2.1.11.3)

The total number of SNMP messages which were delivered to the SNMP entity and were for an unsupported SNMP version.

snmplnBadCommunityNames (1.3.6.1.2.1.11.4)

The total number of community-based SNMP messages (for example, SNMPv1) delivered to the SNMP entity which used an SNMP community name not known to said entity.

Also, implementations which authenticate community-based SNMP messages using check(s) in addition to matching the community name (for example, by also checking whether the message originated from a transport address allowed to use a specified community name) MAY include in this value the number of messages which failed the additional check(s). It is strongly RECOMMENDED that

the documentation for any security model which is used to authenticate community-based SNMP messages specify the precise conditions that contribute to this value.

snmplnBadCommunityUses (1.3.6.1.2.1.11.5)

The total number of community-based SNMP messages (for example, SNMPv1) delivered to the SNMP entity which represented an SNMP operation that was not allowed for the SNMP community named in the message. The precise conditions under which this counter is incremented (if at all) depend on how the SNMP entity implements its access control mechanism and how its applications interact with that access control mechanism. It is strongly RECOMMENDED that the documentation for any access control mechanism which is used to control access to and visibility of MIB instrumentation specify the precise conditions that contribute to this value.

snmplnASNParseErrs (1.3.6.1.2.1.11.6)

The total number of ASN.1 or BER errors encountered by the SNMP entity when decoding received SNMP messages.

snmpEnableAuthenTraps (1.3.6.1.2.1.11.30)

Indicates whether the SNMP entity is permitted to generate authenticationFailure traps. The value of this object overrides any configuration information; as such, it provides a means whereby all authenticationFailure traps may be disabled.

Note that it is strongly recommended that this object be stored in non-volatile memory so that it remains constant across re-initializations of the network management system.

snmpSilentDrops (1.3.6.1.2.1.11.31)

The total number of Confirmed Class PDUs (such as GetRequest-PDUs, GetNextRequest-PDUs, GetBulkRequest-PDUs, SetRequest-PDUs, and InformRequest-PDUs) delivered to the SNMP entity which were silently dropped because the size of a reply containing an alternate Response Class PDU (such as a

Response-PDU) with an empty variable-bindings field was greater than either a local constraint or the maximum message size associated with the originator of the request.

snmpProxyDrops (1.3.6.1.2.1.11.32)

The total number of Confirmed Class PDUs (such as GetRequest-PDUs, GetNextRequest-PDUs, GetBulkRequest-PDUs, SetRequest-PDUs, and InformRequest-PDUs) delivered to the SNMP entity which were silently dropped because the transmission of the (possibly translated) message to a proxy target failed in a manner (other than a time-out) such that no Response Class PDU (such as a Response-PDU) could be returned.

interfaces (1.3.6.1.2.1.2)

ifNumber (1.3.6.1.2.1.2.1)

The number of network interfaces (regardless of their current state) present on this system.

ifTable (1.3.6.1.2.1.2.2)

A list of interface entries. The number of entries is given by the value of ifNumber.

ifIndex (1.3.6.1.2.1.2.2.1.1)

A unique value, greater than zero, for each interface. It is recommended that values are assigned contiguously starting from 1. The value for each interface sub-layer must remain constant at least from one re-initialization of the entity's network management system to the next re-initialization.

ifDescr (1.3.6.1.2.1.2.2.1.2)

A textual string containing information about the interface. This string should include the name of the manufacturer, the product name and the version of the interface hardware/software.

ifType (1.3.6.1.2.1.2.2.1.3)

The type of interface. Additional values for ifType are assigned by the Internet Assigned Numbers Authority (IANA),

through updating the syntax of the IANAifType textual convention.

ifMtu (1.3.6.1.2.1.2.2.1.4)

The size of the largest packet which can be sent/received on the interface, specified in octets. For interfaces that are used for transmitting network datagrams, this is the size of the largest network datagram that can be sent on the interface.

ifSpeed (1.3.6.1.2.1.2.2.1.5)

An estimate of the interface's current bandwidth in bits per second. For interfaces which do not vary in bandwidth or for those where no accurate estimation can be made, this object should contain the nominal bandwidth. If the bandwidth of the interface is greater than the maximum value reportable by this object then this object should report its maximum value (4,294,967,295) and ifHighSpeed must be used to report the interface's speed. For a sub-layer which has no concept of bandwidth, this object should be zero.

ifPhysAddress (1.3.6.1.2.1.2.2.1.6)

The interface's address at its protocol sub-layer. For example, for an 802.x interface, this object normally contains a MAC address. The interface's media-specific MIB must define the bit and byte ordering and the format of the value of this object. For interfaces which do not have such an address (e.g., a serial line), this object should contain an octet string of zero length.

ifAdminStatus (1.3.6.1.2.1.2.2.1.7)

The desired state of the interface. The testing(3) state indicates that no operational packets can be passed. When a managed system initializes, all interfaces start with ifAdminStatus in the down(2) state. As a result of either explicit management action or per configuration information retained by the managed system, ifAdminStatus is then changed to either the up(1) or testing(3) states (or remains in the down(2) state).

ifOperStatus (1.3.6.1.2.1.2.2.1.8)

The current operational state of the interface. The testing(3) state indicates that no operational packets can be passed. If ifAdminStatus is down(2) then ifOperStatus should be down(2). If ifAdminStatus is changed to up(1) then ifOperStatus should change to up(1) if the interface is ready to transmit and receive network traffic; it should change to dormant(5) if the interface is waiting for external actions (such as a serial line waiting for an incoming connection); it should remain in the down(2) state if and only if there is a fault that prevents it from going to the up(1) state; it should remain in the notPresent(6) state if the interface has missing (typically, hardware) components.

ifLastChange (1.3.6.1.2.1.2.2.1.9)

The value of sysUpTime at the time the interface entered its current operational state. If the current state was entered prior to the last re-initialization of the local network management subsystem, then this object contains a zero value.

ifInOctets (1.3.6.1.2.1.2.2.1.10)

The total number of octets received on the interface, including framing characters.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

ifInUcastPkts (1.3.6.1.2.1.2.2.1.11)

The number of packets, delivered by this sub-layer to a higher (sub-)layer, which were not addressed to a multicast or broadcast address at this sub-layer.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

ifInDiscards (1.3.6.1.2.1.2.2.1.13)

The number of inbound packets which were chosen to be

discarded even though no errors had been detected to prevent their being deliverable to a higher-layer protocol. One possible reason for discarding such a packet could be to free up buffer space.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of `ifCounterDiscontinuityTime`.

ifInErrors (1.3.6.1.2.1.2.2.1.14)

For packet-oriented interfaces, the number of inbound packets that contained errors preventing them from being deliverable to a higher-layer protocol. For character-oriented or fixed-length interfaces, the number of inbound transmission units that contained errors preventing them from being deliverable to a higher-layer protocol.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of `ifCounterDiscontinuityTime`.

ifInUnknownProtos (1.3.6.1.2.1.2.2.1.15)

For packet-oriented interfaces, the number of packets received via the interface which were discarded because of an unknown or unsupported protocol. For character-oriented or fixed-length interfaces that support protocol multiplexing the number of transmission units received via the interface which were discarded because of an unknown or unsupported protocol. For any interface that does not support protocol multiplexing, this counter will always be 0.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of `ifCounterDiscontinuityTime`.

ifOutOctets (1.3.6.1.2.1.2.2.1.16)

The total number of octets transmitted out of the interface, including framing characters.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

ifOutUcastPkts (1.3.6.1.2.1.2.2.1.17)

The total number of packets that higher-level protocols requested be transmitted, and which were not addressed to a multicast or broadcast address at this sub-layer, including those that were discarded or not sent.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

ifOutDiscards (1.3.6.1.2.1.2.2.1.19)

The number of outbound packets which were chosen to be discarded even though no errors had been detected to prevent their being transmitted. One possible reason for discarding such a packet could be to free up buffer space.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

ifOutErrors (1.3.6.1.2.1.2.2.1.20)

For packet-oriented interfaces, the number of outbound packets that could not be transmitted because of errors.

For character-oriented or fixed-length interfaces, the number of outbound transmission units that could not be transmitted because of errors.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

ifMIBObjects (1.3.6.1.2.1.31.1)

ifTableLastChange (1.3.6.1.2.1.31.1.5)

The value of sysUpTime at the time of the last creation or deletion of an entry in the ifTable. If the number of

entries has been unchanged since the last re-initialization of the local network management subsystem, then this object contains a zero value.

ifXTable (1.3.6.1.2.1.31.1.1)

A list of interface entries. The number of entries is given by the value of ifNumber. This table contains additional objects for the interface table.

ifInMulticastPkts (1.3.6.1.2.1.31.1.1.2)

The number of packets, delivered by this sub-layer to a higher (sub-)layer, which were addressed to a multicast address at this sub-layer. For a MAC layer protocol, this includes both Group and Functional addresses.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

ifInBroadcastPkts (1.3.6.1.2.1.31.1.1.3)

The number of packets, delivered by this sub-layer to a higher (sub-)layer, which were addressed to a broadcast address at this sub-layer.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

ifOutMulticastPkts (1.3.6.1.2.1.31.1.1.4)

The total number of packets that higher-level protocols requested be transmitted, and which were addressed to a multicast address at this sub-layer, including those that were discarded or not sent. For a MAC layer protocol, this includes both Group and Functional addresses.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

ifOutBroadcastPkts (1.3.6.1.2.1.31.1.1.5)

The total number of packets that higher-level protocols

requested be transmitted, and which were addressed to a broadcast address at this sub-layer, including those that were discarded or not sent.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of `ifCounterDiscontinuityTime`.

ifHCInOctets (1.3.6.1.2.1.31.1.1.1.6)

The total number of octets received on the interface, including framing characters. This object is a 64-bit version of `ifInOctets`.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of `ifCounterDiscontinuityTime`.

ifHCInUcastPkts (1.3.6.1.2.1.31.1.1.1.7)

The number of packets, delivered by this sub-layer to a higher (sub-)layer, which were not addressed to a multicast or broadcast address at this sub-layer. This object is a 64-bit version of `ifInUcastPkts`.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of `ifCounterDiscontinuityTime`.

ifHCInMulticastPkts (1.3.6.1.2.1.31.1.1.1.8)

The number of packets, delivered by this sub-layer to a higher (sub-)layer, which were addressed to a multicast address at this sub-layer. For a MAC layer protocol, this includes both Group and Functional addresses. This object is a 64-bit version of `ifInMulticastPkts`.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of `ifCounterDiscontinuityTime`.

ifHCInBroadcastPkts (1.3.6.1.2.1.31.1.1.1.9)

The number of packets, delivered by this sub-layer to a

higher (sub-)layer, which were addressed to a broadcast address at this sub-layer. This object is a 64-bit version of `ifInBroadcastPkts`.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of `ifCounterDiscontinuityTime`.

ifHCOctets (1.3.6.1.2.1.31.1.1.1.10)

The total number of octets transmitted out of the interface, including framing characters. This object is a 64-bit version of `ifOutOctets`.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of `ifCounterDiscontinuityTime`.

ifHCUcastPkts (1.3.6.1.2.1.31.1.1.1.11)

The total number of packets that higher-level protocols requested be transmitted, and which were not addressed to a multicast or broadcast address at this sub-layer, including those that were discarded or not sent. This object is a 64-bit version of `ifOutUcastPkts`.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of `ifCounterDiscontinuityTime`.

ifHCMulticastPkts (1.3.6.1.2.1.31.1.1.1.12)

The total number of packets that higher-level protocols requested be transmitted, and which were addressed to a multicast address at this sub-layer, including those that were discarded or not sent. For a MAC layer protocol, this includes both Group and Functional addresses. This object is a 64-bit version of `ifOutMulticastPkts`.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of `ifCounterDiscontinuityTime`.

ifHCOutBroadcastPkts (1.3.6.1.2.1.31.1.1.1.13)

The total number of packets that higher-level protocols requested be transmitted, and which were addressed to a broadcast address at this sub-layer, including those that were discarded or not sent. This object is a 64-bit version of ifOutBroadcastPkts.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

ifLinkUpDownTrapEnable (1.3.6.1.2.1.31.1.1.1.14)

Indicates whether linkUp/linkDown traps should be generated for this interface.

By default, this object should have the value enabled(1) for interfaces which do not operate on 'top' of any other interface (as defined in the ifStackTable), and disabled(2) otherwise.

ifHighSpeed (1.3.6.1.2.1.31.1.1.1.15)

An estimate of the interface's current bandwidth in units of 1,000,000 bits per second. If this object reports a value of 'n' then the speed of the interface is somewhere in the range of 'n-500,000' to 'n+499,999'. For interfaces which do not vary in bandwidth or for those where no accurate estimation can be made, this object should contain the nominal bandwidth. For a sub-layer which has no concept of bandwidth, this object should be zero.

ifPromiscuousMode (1.3.6.1.2.1.31.1.1.1.16)

This object has a value of false(2) if this interface only accepts packets/frames that are addressed to this station.

This object has a value of true(1) when the station accepts all packets/frames transmitted on the media. The value true(1) is only legal on certain types of media. If legal, setting this object to a value of true(1) may require the interface to be reset before becoming effective.

The value of ifPromiscuousMode does not affect the reception of broadcast and multicast packets/frames by the interface.

ifConnectorPresent (1.3.6.1.2.1.31.1.1.1.17)

This object has the value 'true(1)' if the interface sublayer has a physical connector and the value 'false(2)' otherwise.

ifAlias (1.3.6.1.2.1.31.1.1.1.18)

This object is an 'alias' name for the interface as specified by a network manager, and provides a non-volatile 'handle' for the interface.

On the first instantiation of an interface, the value of ifAlias associated with that interface is the zero-length string. As and when a value is written into an instance of ifAlias through a network management set operation, then the agent must retain the supplied value in the ifAlias instance associated with the same interface for as long as that interface remains instantiated, including across all re-initializations/reboots of the network management system, including those which result in a change of the interface's ifIndex value.

An example of the value which a network manager might store in this object for a WAN interface is the (Telco's) circuit number/identifier of the interface.

Some agents may support write-access only for interfaces having particular values of ifType. An agent which supports write access to this object is required to keep the value in non-volatile storage, but it may limit the length of new values depending on how much storage is already occupied by the current values for other interfaces.

ifCounterDiscontinuityTime (1.3.6.1.2.1.31.1.1.1.19)

The value of sysUpTime on the most recent occasion at which any one or more of this interface's counters suffered a discontinuity. The relevant counters are the specific instances associated with this interface of any Counter32 or Counter64 object contained in the ifTable or ifXTable. If no such discontinuities have occurred since the last re-initialization of the local management subsystem, then this object contains a zero value.

icmp (1.3.6.1.2.1.5)

udp (1.3.6.1.2.1.7)

udpInDatagrams (1.3.6.1.2.1.7.1)

The total number of UDP datagrams delivered to UDP users.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by discontinuities in the value of sysUpTime.

udpNoPorts (1.3.6.1.2.1.7.2)

The total number of received UDP datagrams for which there was no application at the destination port.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by discontinuities in the value of sysUpTime.

udpInErrors (1.3.6.1.2.1.7.3)

The number of received UDP datagrams that could not be delivered for reasons other than the lack of an application at the destination port.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by discontinuities in the value of sysUpTime.

udpOutDatagrams (1.3.6.1.2.1.7.4)

The total number of UDP datagrams sent from this entity.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by discontinuities in the value of sysUpTime.

NetScaler Enterprise OIDs

nsSysGroup (1.3.6.1.4.1.5951.4.1.1)

sysBuildVersion (1.3.6.1.4.1.5951.4.1.1.1)

This shows the version of the kernel build running on the netscaler.

sysIpAddress (1.3.6.1.4.1.5951.4.1.1.2)

This shows the configured ipaddress of the NetScaler

sysNetmask (1.3.6.1.4.1.5951.4.1.1.3)

This shows the configured netmask of the NetScaler

sysHighAvailabilityMode (1.3.6.1.4.1.5951.4.1.1.6)

This shows whether NetScaler is in standalone mode or whether it is primary or secondary in case of failover mode.

sysGateway (1.3.6.1.4.1.5951.4.1.1.7)

This represents the default gateway configured on the NetScaler

sysCurMappedIpCount (1.3.6.1.4.1.5951.4.1.1.8)

This represents the number of Mapped IPs currently configured on the NetScaler system

sysCustomID (1.3.6.1.4.1.5951.4.1.1.9)

Configurable Identifier for the system

sysHardwareVersionId (1.3.6.1.4.1.5951.4.1.1.10)

The hardware version ID of the NetScaler system

sysHardwareVersionDesc (1.3.6.1.4.1.5951.4.1.1.11)

The hardware version description of the NetScaler system

sysTotConfigChanges (1.3.6.1.4.1.5951.4.1.1.12)

The number of times a configuration change was made on the NetScaler appliance.

sysTotSaveConfigs (1.3.6.1.4.1.5951.4.1.1.13)

Number of times the system configuration was saved on the NetScaler appliance.

sysHardwareSerialNumber (1.3.6.1.4.1.5951.4.1.1.14)

The serial number of the NetScaler system.

sysHardwareEncodedSerialNumber (1.3.6.1.4.1.5951.4.1.1.15)

The encoded serial no of the NetScaler system.

sysModelId (1.3.6.1.4.1.5951.4.1.1.16)

The model ID is populated if the system is such that it is license controlled. If the system does not support license based models, then the model id will be zero.

nsFeatureInfo (1.3.6.1.4.1.5951.4.1.1.20)**featureWebLogging (1.3.6.1.4.1.5951.4.1.1.20.1)**

This represents whether webLogging feature is enabled or disabled on NetScaler.

featureSurgeProtection (1.3.6.1.4.1.5951.4.1.1.20.2)

This represents whether surgeProtection feature is enabled or disabled on NetScaler.

featureLoadBalancing (1.3.6.1.4.1.5951.4.1.1.20.3)

This represents whether LoadBalancing feature is enabled or disabled on NetScaler.

featureContentSwitching (1.3.6.1.4.1.5951.4.1.1.20.4)

This represents whether contentSwitching feature is enabled or disabled on NetScaler.

featureCacheRedirection (1.3.6.1.4.1.5951.4.1.1.20.5)

This represents whether cacheRedirection feature is enabled or disabled on NetScaler.

featureSureConnect (1.3.6.1.4.1.5951.4.1.1.20.6)

This represents whether sureConnect feature is enabled or disabled on NetScaler.

featureCompression (1.3.6.1.4.1.5951.4.1.1.20.7)

This represents whether compression feature is enabled or disabled on NetScaler.

featurePriorityQueuing (1.3.6.1.4.1.5951.4.1.1.20.8)

This represents whether priorityQueuing feature is enabled or disabled on NetScaler.

featureSslOffloading (1.3.6.1.4.1.5951.4.1.1.20.9)

This represents whether sslOffloading feature is enabled or disabled on NetScaler.

featureGslb (1.3.6.1.4.1.5951.4.1.1.20.10)

This represents whether gslb feature is enabled or disabled on NetScaler.

featureHttpDosProtection (1.3.6.1.4.1.5951.4.1.1.20.11)

This represents whether httpDosProtection feature is enabled or disabled on NetScaler.

featureContentFiltering (1.3.6.1.4.1.5951.4.1.1.20.13)

This represents whether contentFiltering feature is enabled or disabled on NetScaler.

featureInternalCaching (1.3.6.1.4.1.5951.4.1.1.20.14)

This represents whether internalCaching feature is enabled or disabled on NetScaler.

featureSSLVPN (1.3.6.1.4.1.5951.4.1.1.20.15)

This represents whether SSL VPN feature is enabled or disabled on NetScaler.

featureOSPF (1.3.6.1.4.1.5951.4.1.1.20.16)

This represents whether OSPF feature is enabled or disabled on NetScaler.

featureRIP (1.3.6.1.4.1.5951.4.1.1.20.17)

This represents whether RIP feature is enabled or disabled on NetScaler.

featureBGP (1.3.6.1.4.1.5951.4.1.1.20.18)

This represents whether BGP feature is enabled or disabled on NetScaler.

featureRewrite (1.3.6.1.4.1.5951.4.1.1.20.19)

This represents whether Rewrite feature is enabled or disabled on NetScaler.

featureDeltaCompression (1.3.6.1.4.1.5951.4.1.1.20.20)

This represents whether Delta Compression feature is enabled or disabled on NetScaler.

featureGSLBProximity (1.3.6.1.4.1.5951.4.1.1.20.21)

This represents whether GSLB Proximity feature is enabled or disabled on NetScaler.

featureIPv6ProtocolTranslation (1.3.6.1.4.1.5951.4.1.1.20.22)

This represents whether IPv6 Protocol Translation feature is enabled or disabled on NetScaler.

featureApplicationFirewall (1.3.6.1.4.1.5951.4.1.1.20.23)

This represents whether Application Firewall feature is enabled or disabled on NetScaler.

featureResponder (1.3.6.1.4.1.5951.4.1.1.20.24)

This represents whether Responder feature is enabled or disabled on NetScaler.

featureHtmlInjection (1.3.6.1.4.1.5951.4.1.1.20.25)

This represents whether Html Injection feature is enabled or disabled on Netscaler.

featureAGEE (1.3.6.1.4.1.5951.4.1.1.20.50)

This represents whether AGEE feature of SSLVPN is enabled or disabled on Netscaler.

featureAAA (1.3.6.1.4.1.5951.4.1.1.20.51)

This represents whether Authentication, Authorization and Auditing features for Traffic Management vservers are enabled or disabled on NetScaler.

featurePLATFORM (1.3.6.1.4.1.5951.4.1.1.20.60)

This gives platform information AGEE 1 NSVA 2 etc

featureAPPFLOW (1.3.6.1.4.1.5951.4.1.1.20.61)

This represents whether APPFLOW feature is enabled or disable on NetScaler.

featureISIS (1.3.6.1.4.1.5951.4.1.1.20.62)

This represents whether ISIS feature is enabled or disabled on NetScaler.

featureContentAdapation (1.3.6.1.4.1.5951.4.1.1.20.63)

This represents whether ContentAccelerator feature is enabled or disabled on NetScaler.

nsModelInfo (1.3.6.1.4.1.5951.4.1.1.21)**modeFastRamp (1.3.6.1.4.1.5951.4.1.1.21.1)**

This represents whether fastRamp mode is enabled or disabled on NetScaler.

I2Mode (1.3.6.1.4.1.5951.4.1.1.21.2)

This represents whether I2Mode mode is enabled or disabled on NetScaler.

modeUseSrcIp (1.3.6.1.4.1.5951.4.1.1.21.3)

This represents whether useSrcIp mode is enabled or disabled on NetScaler.

modeClientKeepAlive (1.3.6.1.4.1.5951.4.1.1.21.4)

This represents whether clientKeepAlive mode is enabled or disabled on NetScaler.

modeTcpBuffering (1.3.6.1.4.1.5951.4.1.1.21.5)

This represents whether tcpBuffering mode is enabled or disabled on NetScaler.

modeMacBasedForwarding (1.3.6.1.4.1.5951.4.1.1.21.6)

This represents whether macBasedForwarding mode is enabled or disabled on NetScaler.

modeUseSubnetIp (1.3.6.1.4.1.5951.4.1.1.21.7)

This represents whether Use Subnet IP mode is enabled or disabled on NetScaler.

modeEdgeConfiguration (1.3.6.1.4.1.5951.4.1.1.21.8)

This represents whether Edge Configuration mode is enabled or disabled on NetScaler.

I3mode (1.3.6.1.4.1.5951.4.1.1.21.9)

This represents whether I3 mode (ip forwarding) is enabled or disabled on NetScaler.

modePathMTUDiscovery (1.3.6.1.4.1.5951.4.1.1.21.10)

This represents whether path MTU discovery mode is enabled or disabled on NetScaler.

modeStaticRouteAdv (1.3.6.1.4.1.5951.4.1.1.21.11)

This represents whether static route advertisement mode is enabled or disabled on NetScaler.

modeDirectRouteAdv (1.3.6.1.4.1.5951.4.1.1.21.12)

This represents whether direct route advertisement mode is enabled or disabled on NetScaler.

modelIntranetRouteAdv (1.3.6.1.4.1.5951.4.1.1.21.13)

This represents whether intranet route advertisement mode is enabled or disabled on NetScaler.

brgBpdu (1.3.6.1.4.1.5951.4.1.1.21.14)

This represents whether Bridging of BPDU is enabled or disabled on NetScaler.

modelIpv6StaticRouteAdv (1.3.6.1.4.1.5951.4.1.1.21.15)

This represents whether Ipv6 static route advertisement mode is enabled or disabled on NetScaler.

modelIpv6DirectRouteAdv (1.3.6.1.4.1.5951.4.1.1.21.16)

This represents whether Ipv6 direct route advertisement mode is enabled or disabled on NetScaler.

nsHighAvailabilityGroup (1.3.6.1.4.1.5951.4.1.1.23)**haPeerId (1.3.6.1.4.1.5951.4.1.1.23.1)**

The unique identifier to represent the failover peer NetScaler

haPeerIpAddress (1.3.6.1.4.1.5951.4.1.1.23.2)

This represents the ipaddress of the failover peer NetScaler(Only for HA over IPv4). For HA over IPv6 (as well as IPv4) haPeerInetAddr will contain this information.

haPeerState (1.3.6.1.4.1.5951.4.1.1.23.3)

This represents the state of the failover peer NetScaler whether Primary or Secondary

haTotStateTransitions (1.3.6.1.4.1.5951.4.1.1.23.4)

Total number of master state changes that the NetScaler appliance has made from primary to secondary and vice-versa.

haTimeofLastStateTransition (1.3.6.1.4.1.5951.4.1.1.23.5)

This represents the time since the NetScaler underwent a state change from primary to secondary or vice-versa

haTotStateFail (1.3.6.1.4.1.5951.4.1.1.23.6)

Number of times state changed to
PARTIAL_FAIL/PARTIAL_FAIL_SSL/ROUTEMONITOR_FAIL/COMPLETE_FAIL.

haErrSyncFailure (1.3.6.1.4.1.5951.4.1.1.23.7)

Number of times the configuration of the primary and secondary nodes failed to synchronize since that last transition. A synchronization failure results in mismatched configuration. It can be caused by a mismatch in the Remote Procedural Call (RPC) password on the two nodes forming the high availability pair.

haErrTotNodeDown (1.3.6.1.4.1.5951.4.1.1.23.8)

Total number of heart-beats missed while the peer node was DOWN.

haErrPropMemFail (1.3.6.1.4.1.5951.4.1.1.23.9)

Total number of times memory allocation failed during command propagation.

haErrNsbMemFail (1.3.6.1.4.1.5951.4.1.1.23.10)

Total number of times memory allocation failed while sending heartbeats.

haErrPortSilent (1.3.6.1.4.1.5951.4.1.1.23.11)

Total number of times heartbeat packets were not received on any enabled interface for the duration of the Dead Interval.

haTotTimerRecoveries (1.3.6.1.4.1.5951.4.1.1.23.12)

Total number of times HA engine recovered from tight loops. (i.e., Total number of times HA timers are not called for MAX down time).

haNicsMonitorFailed (1.3.6.1.4.1.5951.4.1.1.23.14)

Interfaces on which HA heartbeats are not being seen

haLastMasterStateTransitionReason (1.3.6.1.4.1.5951.4.1.1.23.15)

The reason for the last master state transition. This gives the conditions under which this node assumed the current state. The current state is available at the oid sysHighAvailabilityMode.0

haPeerSystemState (1.3.6.1.4.1.5951.4.1.1.23.16)

HA peer system state

haErrPropTimeout (1.3.6.1.4.1.5951.4.1.1.23.17)

Number of times propagation timed out.

haCurDerivedInc (1.3.6.1.4.1.5951.4.1.1.23.18)

Derived incarnation based on IOCTLs received.

haCurPeerInc (1.3.6.1.4.1.5951.4.1.1.23.19)

The peer's incarnation as seen from heartbeats.

haErrMasterDispute (1.3.6.1.4.1.5951.4.1.1.23.20)

Number of HA master disputes.

haTotPktTx (1.3.6.1.4.1.5951.4.1.1.23.21)

Number of heartbeat packets sent to the peer node. Heartbeats are sent at regular intervals (default is 200 milliseconds) to determine the state of the peer node.

haTotPktRx (1.3.6.1.4.1.5951.4.1.1.23.22)

Number of heartbeat packets received from the peer node. Heartbeats are sent at regular intervals (default is 200 milliseconds) to determine the state of the peer node.

haCurStatus (1.3.6.1.4.1.5951.4.1.1.23.23)

Whether a NetScaler appliance is configured for high availability. Possible values are YES and NO. If the value is NO, the high availability statistics below are invalid.

haCurState (1.3.6.1.4.1.5951.4.1.1.23.24)

State of the HA node, based on its health, in a high availability setup. Possible values are:

UP ? Indicates that the node is accessible and can function as either a primary or secondary node.

DISABLED ? Indicates that the high availability status of the node has been manually disabled. Synchronization and propagation cannot take place between the peer nodes.

INIT ? Indicates that the node is in the process of becoming part of the high availability configuration.

PARTIALFAIL ? Indicates that one of the high availability monitored interfaces has failed because of a card or link failure. This state triggers a failover.

COMPLETEFAIL ? Indicates that all the interfaces of the node are unusable, because the interfaces on which high availability monitoring is enabled are not connected or are manually disabled. This state triggers a failover.

DUMB ? Indicates that the node is in listening mode. It does not participate in high availability transitions or transfer configuration from the peer node. This is a configured value, not a statistic.

PARTIALFAILSSL ? Indicates that the SSL card has failed. This state triggers a failover.

ROUTEMONITORFAIL ? Indicates that the route monitor has failed. This state triggers a failover.

haPeerInetAddrType (1.3.6.1.4.1.5951.4.1.1.23.25)

The address type of haPeerInetAddr

haPeerInetAddr (1.3.6.1.4.1.5951.4.1.1.23.26)

This represents the Internet Address of the failover peer NetScaler

haNicMonitorSucceeded (1.3.6.1.4.1.5951.4.1.1.23.27)

Heartbeat succeeded on this nic.

haLastNicMonitorFailed (1.3.6.1.4.1.5951.4.1.1.23.28)

Heartbeat failed on this nic.

vlanTable (1.3.6.1.4.1.5951.4.1.1.24)

The vlan related statistics Table.

Indexed on: [vlanId](#)

vlanId (1.3.6.1.4.1.5951.4.1.1.24.1.1)

This represents the unique id of the vlan

vlanMemberInterfaces (1.3.6.1.4.1.5951.4.1.1.24.1.2)

This represents the list of interfaces on the NetScaler that are members of the vlan

vlanTaggedInterfaces (1.3.6.1.4.1.5951.4.1.1.24.1.3)

This represents the list of interfaces on the NetScaler that are members of the vlan that carry tagged packets

vlanTotRxPkts (1.3.6.1.4.1.5951.4.1.1.24.1.16)

Packets received on the VLAN.

vlanTotRxBytes (1.3.6.1.4.1.5951.4.1.1.24.1.17)

Bytes of data received on the VLAN.

vlanTotTxPkts (1.3.6.1.4.1.5951.4.1.1.24.1.18)

Packets transmitted on the VLAN.

vlanTotTxBytes (1.3.6.1.4.1.5951.4.1.1.24.1.19)

Bytes of data transmitted on the VLAN.

vlanTotDroppedPkts (1.3.6.1.4.1.5951.4.1.1.24.1.20)

Inbound packets dropped by the VLAN upon reception.

vlanTotBroadcastPkts (1.3.6.1.4.1.5951.4.1.1.24.1.21)

Broadcast packets sent and received on the VLAN.

vlanBridgeGroup (1.3.6.1.4.1.5951.4.1.1.24.1.24)

This represents the bridge group to which this vlan is bound.

vlanAliasName (1.3.6.1.4.1.5951.4.1.1.24.1.25)

This is vlan alias name if configured

nsIpAddrTable (1.3.6.1.4.1.5951.4.1.1.26)

This table contains information about the IP addresses configured on the NetScaler.

Indexed on: [ipAddr](#)

ipAddr (1.3.6.1.4.1.5951.4.1.1.26.1.1)

This represents an IP address configured on the NetScaler

ipNetmask (1.3.6.1.4.1.5951.4.1.1.26.1.2)

This represents the Netmask

ipType (1.3.6.1.4.1.5951.4.1.1.26.1.3)

This represents the IP address type

ipMode (1.3.6.1.4.1.5951.4.1.1.26.1.4)

This represents the IP address mode

ipFreePorts (1.3.6.1.4.1.5951.4.1.1.26.1.5)

This represents the number of unused ports free on this IP

ipVlan (1.3.6.1.4.1.5951.4.1.1.26.1.6)

The vlan to which this ip address is bound.

ipBridgeGroup (1.3.6.1.4.1.5951.4.1.1.26.1.7)

The bridge group to which this ip address is bound.

ipVxlan (1.3.6.1.4.1.5951.4.1.1.26.1.8)

The vxlan to which this ip address is bound.

nsResourceGroup (1.3.6.1.4.1.5951.4.1.1.41)

resCpuUsage (1.3.6.1.4.1.5951.4.1.1.41.1)

CPU utilization percentage.

resMemUsage (1.3.6.1.4.1.5951.4.1.1.41.2)

Percentage of memory utilization on NetScaler.

numCPUs (1.3.6.1.4.1.5951.4.1.1.41.3)

The number of active CPUs.

memSizeMB (1.3.6.1.4.1.5951.4.1.1.41.4)

Total amount of system memory, in megabytes.

numSSLCards (1.3.6.1.4.1.5951.4.1.1.41.5)

Number of SSL Cards on the system

nsCPUTable (1.3.6.1.4.1.5951.4.1.1.41.6)

This table contains information about each CPU in NetScaler.

Indexed on: [nsCPUName](#)

nsCPUName (1.3.6.1.4.1.5951.4.1.1.41.6.1.1)

The name of the CPU.

nsCPUUsage (1.3.6.1.4.1.5951.4.1.1.41.6.1.2)

CPU utilization percentage.

nsSysHealthTable (1.3.6.1.4.1.5951.4.1.1.41.7)

This table contains information about the System Health status of the NetScaler.

Indexed on: [sysHealthCounterName](#)

sysHealthCounterName (1.3.6.1.4.1.5951.4.1.1.41.7.1.1)

This is the health counter name. The counter name is composed with the type of the counter and what it is intended to carry. All voltage counters starts with 'v', fan counters starts with 'fan' and temperature counters starts with 't'. Eg: 'v50p' is a voltage counter that carries the value of the +5v counter.

sysHealthCounterValue (1.3.6.1.4.1.5951.4.1.1.41.7.1.2)

The health counters value. The units are 'mv', RPM and degrees Celsius for voltage, fan and temperatures respectively.

nsSysHealthDiskTable (1.3.6.1.4.1.5951.4.1.1.41.8)

This table contains information about the disk space of the NetScaler.

Indexed on: [sysHealthDiskName](#)

sysHealthDiskName (1.3.6.1.4.1.5951.4.1.1.41.8.1.1)

The disk name. Disk name always starts with the 'disk' keyword. Eg: disk0, disk1. Currently disk0 is mapped to /flash and disk1 mapped to /var partitions.

sysHealthDiskSize (1.3.6.1.4.1.5951.4.1.1.41.8.1.2)

The total disk space in MBytes (includes available and used spaces also).

sysHealthDiskAvail (1.3.6.1.4.1.5951.4.1.1.41.8.1.3)

The total disk space available in MBytes.

sysHealthDiskUsed (1.3.6.1.4.1.5951.4.1.1.41.8.1.4)

The total disk space used in MBytes.

sysHealthDiskPerusage (1.3.6.1.4.1.5951.4.1.1.41.8.1.5)

The Percentage of the disk space used.

cpuSpeedMHz (1.3.6.1.4.1.5951.4.1.1.41.15)

CPU speed in MHz.

numPEs (1.3.6.1.4.1.5951.4.1.1.41.16)

The Netscaler number of PEs running

sysStatisticsTime (1.3.6.1.4.1.5951.4.1.1.41.17)

This gives the timestamp of the statistics returned by SNMP counter values. This can be used for calculating the rate of increments for Counter and Counter64 type of OIDs.

nsIpStatsGroup (1.3.6.1.4.1.5951.4.1.1.43)**ipTotRxPkts (1.3.6.1.4.1.5951.4.1.1.43.25)**

IP packets received.

ipTotRxBytes (1.3.6.1.4.1.5951.4.1.1.43.26)

Bytes of IP data received.

ipTotRxMbits (1.3.6.1.4.1.5951.4.1.1.43.27)

Megabits of IP data received.

ipTotTxPkts (1.3.6.1.4.1.5951.4.1.1.43.28)

IP packets transmitted.

ipTotTxBytes (1.3.6.1.4.1.5951.4.1.1.43.29)

Bytes of IP data transmitted.

ipTotTxMbits (1.3.6.1.4.1.5951.4.1.1.43.30)

Megabits of IP data transmitted.

ipTotFragments (1.3.6.1.4.1.5951.4.1.1.43.31)

IP fragments received.

ipTotBadlens (1.3.6.1.4.1.5951.4.1.1.43.32)

Packets received with a length greater than the normal maximum transmission unit of 1514 bytes.

ipTotBadMacAddrs (1.3.6.1.4.1.5951.4.1.1.43.33)

IP packets transmitted with a bad MAC address.

ipTotMaxClients (1.3.6.1.4.1.5951.4.1.1.43.34)

Attempts to open a new connection to a service for which the maximum limit has been exceeded. Default value, 0, applies no limit.

ipTotUnknownSvcs (1.3.6.1.4.1.5951.4.1.1.43.35)

Packets received on a port or service that is not configured.

ipTotLandattacks (1.3.6.1.4.1.5951.4.1.1.43.36)

Land-attack packets received. The source and destination addresses are the same.

ipTotBadChecksums (1.3.6.1.4.1.5951.4.1.1.43.37)

Packets received with an IP checksum error.

ipTotReassemblyAttempt (1.3.6.1.4.1.5951.4.1.1.43.38)

IP packets that the NetScaler attempts to reassemble. If one of the fragments is missing, the whole packet is dropped.

ipTotSuccReassembly (1.3.6.1.4.1.5951.4.1.1.43.39)

Fragmented IP packets successfully reassembled on the NetScaler.

ipTotUnsuccReassembly (1.3.6.1.4.1.5951.4.1.1.43.40)

Packets received that could not be reassembled. This can occur when there is a checksum failure, an identification field mismatch, or when one of the fragments is missing.

ipTotTooBig (1.3.6.1.4.1.5951.4.1.1.43.41)

Packets received for which the reassembled data exceeds the Ethernet packet data length of 1500 bytes.

ipTotZeroFragmentLen (1.3.6.1.4.1.5951.4.1.1.43.42)

Packets received with a fragment length of 0 bytes.

ipTotDupFragments (1.3.6.1.4.1.5951.4.1.1.43.43)

Duplicate IP fragments received. This can occur when the acknowledgement was not received within the expected time.

ipTotOutOfOrderFrag (1.3.6.1.4.1.5951.4.1.1.43.44)

Fragments received that are out of order.

ipTotUnknownDstRcvd (1.3.6.1.4.1.5951.4.1.1.43.45)

Packets received in which the destination IP address was not reachable or not owned by the NetScaler.

ipTotBadTransport (1.3.6.1.4.1.5951.4.1.1.43.46)

Packets received in which the protocol specified in the IP header is unknown to the NetScaler.

ipTotVIPDown (1.3.6.1.4.1.5951.4.1.1.43.47)

Packets received for which the VIP is down. This can occur when all the services bound to the VIP are down or the VIP is manually disabled.

ipTotFixHeaderFail (1.3.6.1.4.1.5951.4.1.1.43.48)

Packets received that contain an error in one or more components of the IP header.

ipTotAddrLookup (1.3.6.1.4.1.5951.4.1.1.43.49)

IP address lookups performed by the NetScaler. When a packet is received on a non-established session, the NetScaler checks if the destination IP address is one of the NetScaler owned IP addresses.

ipTotAddrLookupFail (1.3.6.1.4.1.5951.4.1.1.43.50)

IP address lookups performed by the NetScaler that have failed because the destination IP address of the packet does not match any of the NetScaler owned IP addresses.

ipTotUDPfragmentsFwd (1.3.6.1.4.1.5951.4.1.1.43.51)

UDP fragments forwarded to the client or the server.

ipTotTCPfragmentsFwd (1.3.6.1.4.1.5951.4.1.1.43.52)

TCP fragments forwarded to the client or the server.

ipTotFragPktsGen (1.3.6.1.4.1.5951.4.1.1.43.53)

Fragmented packets created by the NetScaler.

ipTotInvalidHeaderSz (1.3.6.1.4.1.5951.4.1.1.43.54)

Packets received in which an invalid data length is specified, or the value in the length field and the actual data length do not match. The range for the Ethernet packet data length is 0-1500 bytes.

ipTotInvalidPacketSize (1.3.6.1.4.1.5951.4.1.1.43.55)

Total number of packets received by NetScaler with invalid IP packet size.

ipTotTruncatedPackets (1.3.6.1.4.1.5951.4.1.1.43.56)

Truncated IP packets received. An overflow in the routers along the path can truncate IP packets.

ipTotZeroNextHop (1.3.6.1.4.1.5951.4.1.1.43.57)

Packets received that contain a 0 value in the next hop field. These packets are dropped.

ipTotTtlExpired (1.3.6.1.4.1.5951.4.1.1.43.58)

Packets for which the time-to-live (TTL) expired during transit. These packets are dropped.

nonIpTotTruncatedPackets (1.3.6.1.4.1.5951.4.1.1.43.59)

Truncated non-IP packets received.

nsIcmpStatsGroup (1.3.6.1.4.1.5951.4.1.1.44)

icmpCurRateThreshold (1.3.6.1.4.1.5951.4.1.1.44.17)

Limit for ICMP packets handled every 10 milliseconds. Default value, 0, applies no limit.

This is a configurable value using the set rateControl command.

icmpTotRxPkts (1.3.6.1.4.1.5951.4.1.1.44.22)

ICMP packets received.

icmpTotRxBytes (1.3.6.1.4.1.5951.4.1.1.44.23)

Bytes of ICMP data received.

icmpTotTxPkts (1.3.6.1.4.1.5951.4.1.1.44.24)

ICMP packets transmitted.

icmpTotTxBytes (1.3.6.1.4.1.5951.4.1.1.44.25)

Bytes of ICMP data transmitted.

icmpTotRxEchoReply (1.3.6.1.4.1.5951.4.1.1.44.26)

ICMP Ping echo replies received.

icmpTotTxEchoReply (1.3.6.1.4.1.5951.4.1.1.44.27)

ICMP Ping echo replies transmitted.

icmpTotRxEcho (1.3.6.1.4.1.5951.4.1.1.44.28)

ICMP Ping Echo Request and Echo Reply packets received.

icmpTotPktsDropped (1.3.6.1.4.1.5951.4.1.1.44.29)

ICMP packets dropped because the rate threshold has been exceeded.

icmpTotThresholdExceeds (1.3.6.1.4.1.5951.4.1.1.44.30)

Times the ICMP rate threshold is exceeded. If this counter continuously increases, first make sure the ICMP packets received are genuine. If they are, increase the current rate threshold.

icmpTotPortUnreachableRx (1.3.6.1.4.1.5951.4.1.1.44.31)

ICMP Port Unreachable error messages received. This error is generated when there is no service is running on the port.

icmpTotPortUnreachableTx (1.3.6.1.4.1.5951.4.1.1.44.32)

ICMP Port Unreachable error messages generated. This error is generated when there is no service is running on the port.

icmpTotBadChecksum (1.3.6.1.4.1.5951.4.1.1.44.33)

ICMP Fragmentation Needed error messages received with an ICMP checksum error.

icmpTotNeedFragRx (1.3.6.1.4.1.5951.4.1.1.44.34)

ICMP Fragmentation Needed error messages received for packets that need to be fragmented but for which Don't Fragment is specified the header.

icmpTotNonFirstIpFrag (1.3.6.1.4.1.5951.4.1.1.44.35)

ICMP Fragmentation Needed error messages received that were generated by an IP fragment other than the first one.

icmpTotInvalidBodyLen (1.3.6.1.4.1.5951.4.1.1.44.36)

ICMP Fragmentation Needed error messages received that specified an invalid body length.

icmpTotNoTcpConn (1.3.6.1.4.1.5951.4.1.1.44.37)

ICMP Need Fragmentation error messages received for TCP packets. The state of the connection for these packets is not maintained on the NetScaler.

icmpTotNoUdpConn (1.3.6.1.4.1.5951.4.1.1.44.38)

ICMP Need Fragmentation error messages received for UDP packets. The state of the connection for these packets is not maintained on the NetScaler.

icmpTotInvalidTcpSeqno (1.3.6.1.4.1.5951.4.1.1.44.39)

ICMP Fragmentation Needed error messages received for packets that contain an invalid TCP address.

icmpTotInvalidNextMTUval (1.3.6.1.4.1.5951.4.1.1.44.40)

ICMP Fragmentation Needed error messages received in which the Maximum Transmission Unit (MTU) for the next hop is out of range. The range for the MTU is 576-1500.

icmpTotDstIpLookup (1.3.6.1.4.1.5951.4.1.1.44.41)

Total number of MTU lookup on destination IP info received on a need fragmentation ICMP error message failed.

icmpTotBigNextMTU (1.3.6.1.4.1.5951.4.1.1.44.42)

ICMP Fragmentation Needed error messages received in which the value for the next MTU is higher than that of the current MTU.

icmpTotInvalidProtocol (1.3.6.1.4.1.5951.4.1.1.44.43)

ICMP Fragmentation Needed error messages received that contain a protocol other than TCP and UDP.

icmpTotBadPMTUIpChecksum (1.3.6.1.4.1.5951.4.1.1.44.44)

ICMP Fragmentation Needed error messages received with an IP checksum error.

icmpTotPMTUNoLink (1.3.6.1.4.1.5951.4.1.1.44.45)

ICMP Fragmentation Needed error messages received on a Protocol Control Block (PCB) with no link. The PCB maintains the state of the connection.

icmpTotPMTUDiscoveryDisabled (1.3.6.1.4.1.5951.4.1.1.44.46)

ICMP Need Fragmentation error messages received when the PMTU Discovery mode is not enabled.

nsUdpStatsGroup (1.3.6.1.4.1.5951.4.1.1.45)**udpCurRateThreshold (1.3.6.1.4.1.5951.4.1.1.45.11)**

Limit for UDP packets handled every 10 milliseconds. Default value, 0, applies no limit.

This is a configurable value using the set rateControl command.

udpTotUnknownSvcPkts (1.3.6.1.4.1.5951.4.1.1.45.16)

Stray UDP packets dropped due to no configured listening service.

udpTotRxPkts (1.3.6.1.4.1.5951.4.1.1.45.17)

Total number of UDP packets received.

udpTotRxBytes (1.3.6.1.4.1.5951.4.1.1.45.18)

Total number of UDP data received in bytes.

udpTotTxPkts (1.3.6.1.4.1.5951.4.1.1.45.19)

Total number of UDP packets transmitted.

udpTotTxBytes (1.3.6.1.4.1.5951.4.1.1.45.20)

Total number of UDP data transmitted in bytes.

udpCurRateThresholdExceeds (1.3.6.1.4.1.5951.4.1.1.45.21)

Number of times the UDP rate threshold is exceeded. If this counter continuously increases, first make sure the UDP packets received are genuine.

If they are, increase the current rate threshold. This is a configurable value using the set rateControl command.

udpBadChecksum (1.3.6.1.4.1.5951.4.1.1.45.22)

Packets received with a UDP checksum error.

nsTcpStatsGroup (1.3.6.1.4.1.5951.4.1.1.46)

tcpCurServerConn (1.3.6.1.4.1.5951.4.1.1.46.1)

Server connections, including connections in the Opening, Established, and Closing state.

tcpCurClientConn (1.3.6.1.4.1.5951.4.1.1.46.2)

Client connections, including connections in the Opening, Established, and Closing state.

tcpActiveServerConn (1.3.6.1.4.1.5951.4.1.1.46.8)

Connections to a server currently responding to requests.

tcpCurClientConnClosing (1.3.6.1.4.1.5951.4.1.1.46.9)

Client connections in the Closing state, which indicates that the connection termination process has initiated but is not complete.

tcpCurServerConnEstablished (1.3.6.1.4.1.5951.4.1.1.46.10)

Current server connections in the Established state, which indicates that data transfer can occur between the NetScaler and the server.

tcpCurClientConnOpening (1.3.6.1.4.1.5951.4.1.1.46.11)

Client connections in the Opening state, which indicates that the handshakes are not yet complete.

tcpCurClientConnEstablished (1.3.6.1.4.1.5951.4.1.1.46.12)

Current client connections in the Established state, which indicates that data transfer can occur between the NetScaler and the client.

tcpCurServerConnClosing (1.3.6.1.4.1.5951.4.1.1.46.13)

Server connections in the Closing state, which indicates that the connection termination process has initiated but is not complete.

tcpSpareConn (1.3.6.1.4.1.5951.4.1.1.46.14)

Spare connections available. To save time and resources in establishing another connection for a new client, the connection on the server is not closed after completing the request from the first client and is available for serving future requests.

tcpSurgeQueueLen (1.3.6.1.4.1.5951.4.1.1.46.15)

Connections in the surge queue. When the NetScaler cannot open a connection to the server, for example when maximum connections have been reached, the NetScaler queues these requests.

tcpCurServerConnOpening (1.3.6.1.4.1.5951.4.1.1.46.16)

Server connections in the Opening state, which indicates that the handshakes are not yet complete.

tcpTotServerConnOpened (1.3.6.1.4.1.5951.4.1.1.46.17)

Server connections initiated by the NetScaler since startup. This counter is reset when the NetScaler is restarted.

tcpTotServerConnClosed (1.3.6.1.4.1.5951.4.1.1.46.18)

Total number of closed server connections

tcpTotClientConnOpened (1.3.6.1.4.1.5951.4.1.1.46.19)

Client connections opened by the NetScaler since startup (after three-way handshake). This counter is reset when the NetScaler is restarted.

tcpTotClientConnClosed (1.3.6.1.4.1.5951.4.1.1.46.20)

Total number of closed client connections

tcpTotSyn (1.3.6.1.4.1.5951.4.1.1.46.21)

SYN packets received

tcpTotSynProbe (1.3.6.1.4.1.5951.4.1.1.46.22)

Probes from the NetScaler to a server. The NetScaler sends a SYN packet to the server to check its availability and expects a SYN_ACK packet from the server before a specified response timeout.

tcpTotSvrFin (1.3.6.1.4.1.5951.4.1.1.46.23)

FIN packets received from the server.

tcpTotClntFin (1.3.6.1.4.1.5951.4.1.1.46.24)

FIN packets received from the clients.

tcpWaitToSyn (1.3.6.1.4.1.5951.4.1.1.46.25)

SYN packets (packets used to initiate a TCP connection) received on connections that are in the TIME_WAIT state. Packets cannot be transferred on a connection in this state.

tcpTotZombieClntConnFlushed (1.3.6.1.4.1.5951.4.1.1.46.26)

Client connections that are flushed because the client has been idle for some time.

tcpTotZombieSvrConnFlushed (1.3.6.1.4.1.5951.4.1.1.46.27)

Server connections that are flushed because there have been no client requests in the queue for some time.

tcpTotZombieHalfOpenClntConnFlushed (1.3.6.1.4.1.5951.4.1.1.46.28)

Half-opened client connections that are flushed because the three-way handshakes are not complete.

tcpTotZombieHalfOpenSvrConnFlushed (1.3.6.1.4.1.5951.4.1.1.46.29)

Half-opened server connections that are flushed because the three-way handshakes are not complete.

tcpTotZombieActiveHalfCloseClntConnFlushed (1.3.6.1.4.1.5951.4.1.1.46.30)

Active half-closed client connections that are flushed because the client has closed the connection and there has been no activity on the connection.

tcpTotZombieActiveHalfCloseSvrConnFlushed (1.3.6.1.4.1.5951.4.1.1.46.31)

Active half-closed server connections that are flushed because the server has closed the connection and there has been no activity on the connection.

tcpTotZombiePassiveHalfCloseClntConnFlushed (1.3.6.1.4.1.5951.4.1.1.46.32)

Passive half-closed client connections that are flushed because the NetScaler has closed the connection and there has been no activity on the connection.

tcpTotZombiePassiveHalfCloseSrvConnFlushed (1.3.6.1.4.1.5951.4.1.1.46.33)

Passive half-closed server connections that are flushed because the NetScaler has closed the connection and there has been no activity on the connection.

tcpErrBadChecksum (1.3.6.1.4.1.5951.4.1.1.46.34)

Packets received with a TCP checksum error.

tcpErrSynInSynRcvd (1.3.6.1.4.1.5951.4.1.1.46.35)

SYN packets received on a connection that is in the SYN_RCVD state. A connection goes into the SYN_RCVD state after receiving a SYN packet.

tcpErrSynInEst (1.3.6.1.4.1.5951.4.1.1.46.36)

SYN packets received on a connection that is in the ESTABLISHED state. A SYN packet is not expected on an ESTABLISHED connection.

tcpErrSynGiveUp (1.3.6.1.4.1.5951.4.1.1.46.37)

Attempts to establish a connection on the NetScaler that timed out.

tcpErrSynSentBadAck (1.3.6.1.4.1.5951.4.1.1.46.38)

Incorrect ACK packets received on a connection that is in the SYN_SENT state. An incorrect ACK packet is the third packet in the three-way handshake that has an incorrect sequence number.

tcpErrSynRetry (1.3.6.1.4.1.5951.4.1.1.46.39)

SYN packets resent to a server.

tcpErrFinRetry (1.3.6.1.4.1.5951.4.1.1.46.40)

FIN packets resent to a server or a client.

tcpErrFinGiveUp (1.3.6.1.4.1.5951.4.1.1.46.41)

Connections that were timed out by the NetScaler because of not receiving the ACK packet after retransmitting the FIN packet four times.

tcpErrFinDup (1.3.6.1.4.1.5951.4.1.1.46.42)

Number of duplicate FIN packets received

tcpErrRst (1.3.6.1.4.1.5951.4.1.1.46.43)

Reset packets received from a client or a server.

tcpErrRstNonEst (1.3.6.1.4.1.5951.4.1.1.46.44)

Reset packets received on a connection that is not in the ESTABLISHED state.

tcpErrRstOutOfWindow (1.3.6.1.4.1.5951.4.1.1.46.45)

Reset packets received on a connection that is out of the current TCP window.

tcpErrRstInTimewait (1.3.6.1.4.1.5951.4.1.1.46.46)

Reset packets received on a connection that is in the TIME_WAIT state. Packets cannot be transferred on a connection in the TIME_WAIT state.

tcpErrSvrRetrasmit (1.3.6.1.4.1.5951.4.1.1.46.47)

Packets retransmitted by a server. This usually occurs because the acknowledgement from the NetScaler has not reached the server.

tcpErrCltRetrasmit (1.3.6.1.4.1.5951.4.1.1.46.48)

Packets retransmitted by a client. This usually occurs because the acknowledgement from the NetScaler has not reached the client.

tcpErrFullRetrasmit (1.3.6.1.4.1.5951.4.1.1.46.49)

Full packets retransmitted by the client or the server.

tcpErrPartialRetrasmit (1.3.6.1.4.1.5951.4.1.1.46.50)

Partial packet retransmits by a client or server due to congestion on the connection. This usually occurs because the window advertised by the NetScaler is not big enough to hold the full packet.

tcpErrSvrOutOfOrder (1.3.6.1.4.1.5951.4.1.1.46.51)

Out of order TCP packets received from a server.

tcpErrCltOutOfOrder (1.3.6.1.4.1.5951.4.1.1.46.52)

Out of order TCP packets received from a client.

tcpErrCltHole (1.3.6.1.4.1.5951.4.1.1.46.53)

TCP holes created on a client connection. When out of order packets are received from a client, a hole is created on the NetScaler for each group of missing packets.

tcpErrSvrHole (1.3.6.1.4.1.5951.4.1.1.46.54)

TCP holes created on a server connection. When out of order packets are received from a server, a hole is created on the NetScaler for each group of missing packets.

tcpErrCookiePktSeqReject (1.3.6.1.4.1.5951.4.1.1.46.55)

SYN cookie packets rejected because they contain an incorrect sequence number.

tcpErrCookiePktSigReject (1.3.6.1.4.1.5951.4.1.1.46.56)

SYN cookie packets rejected because they contain an incorrect signature.

tcpErrCookiePktSeqDrop (1.3.6.1.4.1.5951.4.1.1.46.57)

SYN cookie packets dropped because the sequence number specified in the packets is outside the current window.

tcpErrCookiePktMssReject (1.3.6.1.4.1.5951.4.1.1.46.58)

SYN cookie packets rejected because the maximum segment size (MSS) specified in the packets is incorrect.

tcpErrRetransmit (1.3.6.1.4.1.5951.4.1.1.46.59)

TCP packets retransmitted. The NetScaler attempts to retransmit the packet up to seven times, after which it resets the other half of the TCP connection.

tcpErrRetransmitGiveUp (1.3.6.1.4.1.5951.4.1.1.46.60)

Number of times NetScaler terminates a connection after retransmitting the packet seven times on that connection. Retransmission happens when receiving end doesn't acknowledge the packet.

tcpTotRxPkts (1.3.6.1.4.1.5951.4.1.1.46.61)

TCP packets received.

tcpTotRxBytes (1.3.6.1.4.1.5951.4.1.1.46.62)

Bytes of TCP data received.

tcpTotTxPkts (1.3.6.1.4.1.5951.4.1.1.46.63)

TCP packets transmitted.

tcpTotTxBytes (1.3.6.1.4.1.5951.4.1.1.46.64)

Bytes of TCP data transmitted.

pcbTotZombieCall (1.3.6.1.4.1.5951.4.1.1.46.65)

Times the Zombie cleanup function is called. Every time a connection is flushed, it is marked for cleanup. The Zombie cleanup function clears all these connections at predefined intervals.

tcpTotSynHeld (1.3.6.1.4.1.5951.4.1.1.46.66)

SYN packets held on the NetScaler that are waiting for a server connection.

tcpTotSynFlush (1.3.6.1.4.1.5951.4.1.1.46.67)

SYN packets flushed on the NetScaler because of no response from the server for three or more seconds.

tcpTotFinWaitClosed (1.3.6.1.4.1.5951.4.1.1.46.68)

Connections closed on the NetScaler because the number of connections in the TIME_WAIT state has exceeded the default value of 7000.

tcpErrAnyPortFail (1.3.6.1.4.1.5951.4.1.1.46.69)

Port allocations that have failed on a mapped IP address because the maximum limit of 65536 has been exceeded.

tcpErrIpPortFail (1.3.6.1.4.1.5951.4.1.1.46.70)

Port allocations that have failed on a subnet IP address or vserver IP address because the maximum limit of 65536 has been exceeded.

tcpErrSentRst (1.3.6.1.4.1.5951.4.1.1.46.71)

Reset packets sent to a client or a server.

tcpErrBadStateConn (1.3.6.1.4.1.5951.4.1.1.46.72)

Connections that are not in a valid TCP state.

tcpErrFastRetransmissions (1.3.6.1.4.1.5951.4.1.1.46.73)

TCP packets on which the NetScaler performs a fast retransmission in response to three duplicate acknowledgements or a partial acknowledgement. The NetScaler assumes that the packet is lost and retransmits the packet before its time-out.

tcpErrFirstRetransmissions (1.3.6.1.4.1.5951.4.1.1.46.74)

Packets retransmitted once by the NetScaler.

tcpErrSecondRetransmissions (1.3.6.1.4.1.5951.4.1.1.46.75)

Packets retransmitted twice by the NetScaler.

tcpErrThirdRetransmissions (1.3.6.1.4.1.5951.4.1.1.46.76)

Packets retransmitted three times by the NetScaler.

tcpErrForthRetransmissions (1.3.6.1.4.1.5951.4.1.1.46.77)

Packets retransmitted four times by the NetScaler.

tcpErrFifthRetransmissions (1.3.6.1.4.1.5951.4.1.1.46.78)

Packets retransmitted five times by the NetScaler.

tcpErrSixthRetransmissions (1.3.6.1.4.1.5951.4.1.1.46.79)

Packets retransmitted six times by the NetScaler.

tcpErrSeventhRetransmissions (1.3.6.1.4.1.5951.4.1.1.46.80)

Packets retransmitted seven times by the NetScaler. If this fails, the NetScaler terminates the connection.

tcpErrDataAfterFin (1.3.6.1.4.1.5951.4.1.1.46.81)

Packets received following a connection termination request. This error is usually caused by a reordering of packets during transmission.

tcpErrRstThreshold (1.3.6.1.4.1.5951.4.1.1.46.82)

Reset packets dropped because the default threshold of 100 resets per 10 milliseconds has been exceeded. This is a configurable value using the set rateControl command.

tcpErrOutOfWindowPkts (1.3.6.1.4.1.5951.4.1.1.46.83)

Packets received that are out of the current advertised window.

tcpErrSynDroppedCongestion (1.3.6.1.4.1.5951.4.1.1.46.84)

SYN packets dropped because of network congestion.

tcpCurPhysicalServers (1.3.6.1.4.1.5951.4.1.1.46.85)

The number of physical servers that Netscaler has open connections with.

tcpReuseHit (1.3.6.1.4.1.5951.4.1.1.46.86)

Total number of client transactions found the server connection in the reuse-pool.

tcpWaitToData (1.3.6.1.4.1.5951.4.1.1.46.87)

Bytes of data received on connections that are in the TIME_WAIT state. Data cannot be transferred on a connection that is in this state.

tcpErrStrayPkt (1.3.6.1.4.1.5951.4.1.1.46.88)

Number of stray or misrouted packets.

tcpTotClientConnOpenRate (1.3.6.1.4.1.5951.4.1.1.46.89)

Rate at which connections are opened in the system per second.

tcpCurRateThreshold (1.3.6.1.4.1.5951.4.1.1.46.90)

Current threshold for TCP rate control. By default, there is no rate control for TCP.

freeConnHalfClosed (1.3.6.1.4.1.5951.4.1.1.46.91)

Number of half-closed connections that were freed.

freeConnFlushMarked (1.3.6.1.4.1.5951.4.1.1.46.92)

Number of connections freed that were already marked for flush.

freeConnEstd (1.3.6.1.4.1.5951.4.1.1.46.93)

Number of established and active connections freed.

flushThresReached (1.3.6.1.4.1.5951.4.1.1.46.94)

Number of times we reached connection flush threshold.

memFailFlushTrigger (1.3.6.1.4.1.5951.4.1.1.46.95)

Number of flushes triggered through memory failure.

mptcpCurMpcapableSessions (1.3.6.1.4.1.5951.4.1.1.46.96)

The number of current mptcp sessions.

mptcpCurSFConnections (1.3.6.1.4.1.5951.4.1.1.46.97)

The number of current mptcp subflow connections.

mptcpCurPendingJoin (1.3.6.1.4.1.5951.4.1.1.46.98)

The number of current mptcp subflow connections in pending state.

mptcpErrInvalCookie (1.3.6.1.4.1.5951.4.1.1.46.99)

MPTCP invalid cookie received on mp capable final ack.

mptcpErrUnknownToken (1.3.6.1.4.1.5951.4.1.1.46.100)

MPTCP invalid token received on mp join request.

mptcpErrAddridExist (1.3.6.1.4.1.5951.4.1.1.46.101)

MPTCP Mp join request on existing address id.

mptcpErrMaxSF (1.3.6.1.4.1.5951.4.1.1.46.102)

MPTCP new mp join request after maximum configured subflows are established.

mptcpErrInvalMAC (1.3.6.1.4.1.5951.4.1.1.46.103)

MPTCP invalid MAC on mp join final ack.

mptcpErrBadCksum (1.3.6.1.4.1.5951.4.1.1.46.104)

MPTCP checksum failed. Connection will fallback to regular tcp.

mptcpErrAddrId0 (1.3.6.1.4.1.5951.4.1.1.46.105)

MPTCP Mp join request on address id 0.

mptcpErrfastclose (1.3.6.1.4.1.5951.4.1.1.46.106)

MPTCP FAST CLOSE sent.

mptcpErrJoinThreshold (1.3.6.1.4.1.5951.4.1.1.46.107)

MPTCP Global pending mp join threshold limit is reached, new mp join request will be dropped sending RST

mptcpErrInvalOpts (1.3.6.1.4.1.5951.4.1.1.46.108)

MPTCP invalid mptcp option is received and is dropped.

mptcpErrInvalRemAddr (1.3.6.1.4.1.5951.4.1.1.46.109)

MPTCP remove address request received on invalid/unknown address id.

mptcpErrVersionNotSupported (1.3.6.1.4.1.5951.4.1.1.46.110)

MPTCP Mp capable request from unsupported mptcp client.

mptcpErrCryptoNotSupported (1.3.6.1.4.1.5951.4.1.1.46.111)

MPTCP client crypto algorithm not supported.

mptcpErrExtnFlagSet (1.3.6.1.4.1.5951.4.1.1.46.112)

MPTCP Mp capable extension flag is set on mp capable request.

mptcpErrResFlagSet (1.3.6.1.4.1.5951.4.1.1.46.113)

MPTCP One or more reserved bits are set on mp capable request.

mptcpErrJoinAfterFallback (1.3.6.1.4.1.5951.4.1.1.46.114)

MPTCP New join request received after fallback to regular tcp.

mptcpErrDataFinpassive (1.3.6.1.4.1.5951.4.1.1.46.115)

MPTCP Data FIN received on passive subflow

mptcpErrFastClosepassive (1.3.6.1.4.1.5951.4.1.1.46.116)

MPTCP Fast close received on passive subflow.

mptcpErrFastClose (1.3.6.1.4.1.5951.4.1.1.46.117)

MPTCP Fast close received on a subflow.

mptcpErrFastCloseKey (1.3.6.1.4.1.5951.4.1.1.46.118)

MPTCP Fast close received with invalid key and the packet is dropped.

mptcpPlainackFallback (1.3.6.1.4.1.5951.4.1.1.46.119)

MPTCP Fallback to regular tcp on receiving plain ack for DSS.

mptcpPlainackRST (1.3.6.1.4.1.5951.4.1.1.46.120)

MPTCP Sent RST on receiving plain ack for DSS.

mptcpMPFailSent (1.3.6.1.4.1.5951.4.1.1.46.121)

MPTCP Total mp fail sent due to checksum failure.

mptcpMPFailRecvd (1.3.6.1.4.1.5951.4.1.1.46.122)

MPTCP Total mpfail received and fallback to regular tcp.

mptcpInfiniteMapRecvd (1.3.6.1.4.1.5951.4.1.1.46.123)

MPTCP Received and set infinite map and fallen back to regular tcp.

mptcpTotMpCapSyn (1.3.6.1.4.1.5951.4.1.1.46.124)

MPTCP total mpcapable syn received

mptcpTotMpJoinSyn (1.3.6.1.4.1.5951.4.1.1.46.125)

MPTCP total mpjoin syn received

mptcpTotMpcapSession (1.3.6.1.4.1.5951.4.1.1.46.126)

MPTCP total mpcapable session created

mptcpTotSFConn (1.3.6.1.4.1.5951.4.1.1.46.127)

MPTCP total mpjoin connections created

mptcpTotEstSFReplaced (1.3.6.1.4.1.5951.4.1.1.46.128)

MPTCP Total established subflows replaced due to new join.

mptcpTotPendSFReplaced (1.3.6.1.4.1.5951.4.1.1.46.129)

MPTCP Total pending subflows replaced due to new join.

nsSslStatsGroup (1.3.6.1.4.1.5951.4.1.1.47)

sslCardStatus (1.3.6.1.4.1.5951.4.1.1.47.1)

Status of the SSL card(s). The value should be interpreted in binary form, with each set bit indicates a card as UP.

sslEngineStatus (1.3.6.1.4.1.5951.4.1.1.47.2)

State of the SSL Engine (1=UP/0=DOWN). This state is decided based on SSL Feature/License status and minimum number of cards UP.

sslSessionsPerSec (1.3.6.1.4.1.5951.4.1.1.47.3)

SSL sessions per second between client and NetScaler appliance.

sslTotTransactions (1.3.6.1.4.1.5951.4.1.1.47.200)

Number of SSL transactions on the NetScaler appliance.

sslTotSSLv2Transactions (1.3.6.1.4.1.5951.4.1.1.47.201)

Number of SSLv2 transactions on the NetScaler appliance.

sslTotSSLv3Transactions (1.3.6.1.4.1.5951.4.1.1.47.202)

Total number of SSLv3 transactions on the NetScaler appliance.

sslTotTLSv1Transactions (1.3.6.1.4.1.5951.4.1.1.47.203)

Number of TLSv1 transactions on the NetScaler appliance.

sslTotSessions (1.3.6.1.4.1.5951.4.1.1.47.204)

Number of SSL sessions on the NetScaler appliance.

sslTotSSLv2Sessions (1.3.6.1.4.1.5951.4.1.1.47.205)

Number of SSLv2 sessions on the NetScaler appliance.

sslTotSSLv3Sessions (1.3.6.1.4.1.5951.4.1.1.47.206)

Number of SSLv3 sessions on the NetScaler appliance.

sslTotTLSv1Sessions (1.3.6.1.4.1.5951.4.1.1.47.207)

Number of TLSv1 sessions on the NetScaler appliance.

sslTotExpiredSessions (1.3.6.1.4.1.5951.4.1.1.47.208)

Total number of expired SSL sessions on the NetScaler appliance.

sslTotNewSessions (1.3.6.1.4.1.5951.4.1.1.47.209)

Number of new SSL sessions created on the NetScaler appliance.

sslTotSessionHits (1.3.6.1.4.1.5951.4.1.1.47.210)

Number of SSL session reuse hits on the NetScaler appliance.

sslTotSessionMiss (1.3.6.1.4.1.5951.4.1.1.47.211)

Number of SSL session reuse misses on the NetScaler appliance.

sslTotRenegSessions (1.3.6.1.4.1.5951.4.1.1.47.212)

Number of SSL session renegotiations on the NetScaler appliance.

sslTotSSLv3RenegSessions (1.3.6.1.4.1.5951.4.1.1.47.213)

Number of session renegotiations done on SSLv3.

sslTotTLSv1RenegSessions (1.3.6.1.4.1.5951.4.1.1.47.214)

Number of SSL session renegotiations done on TLSv1.

sslTotSSLv2Handshakes (1.3.6.1.4.1.5951.4.1.1.47.215)

Number of handshakes on SSLv2 on the NetScaler appliance.

sslTotSSLv3Handshakes (1.3.6.1.4.1.5951.4.1.1.47.216)

Number of handshakes on SSLv3 on the NetScaler appliance.

sslTotTLSv1Handshakes (1.3.6.1.4.1.5951.4.1.1.47.217)

Number of SSL handshakes on TLSv1 on the NetScaler appliance.

sslTotSSLv2ClientAuthentications (1.3.6.1.4.1.5951.4.1.1.47.218)

Number of client authentications done on SSLv2.

sslTotSSLv3ClientAuthentications (1.3.6.1.4.1.5951.4.1.1.47.219)

Number of client authentications done on SSLv3.

sslTotTLSv1ClientAuthentications (1.3.6.1.4.1.5951.4.1.1.47.220)

Number of client authentications done on TLSv1.

sslTotRSA512keyExchanges (1.3.6.1.4.1.5951.4.1.1.47.221)

Number of RSA 512-bit key exchanges on the NetScaler appliance.

sslTotRSA1024keyExchanges (1.3.6.1.4.1.5951.4.1.1.47.222)

Number of RSA 1024-bit key exchanges on the NetScaler appliance.

sslTotRSA2048keyExchanges (1.3.6.1.4.1.5951.4.1.1.47.223)

Number of RSA 2048-bit key exchanges on the NetScaler appliance.

sslTotDH512keyExchanges (1.3.6.1.4.1.5951.4.1.1.47.224)

Number of Diffie-Helman 512-bit key exchanges on the NetScaler appliance.

sslTotDH1024keyExchanges (1.3.6.1.4.1.5951.4.1.1.47.225)

Number of Diffie-Helman 1024-bit key exchanges on the NetScaler appliance.

sslTotDH2048keyExchanges (1.3.6.1.4.1.5951.4.1.1.47.226)

Number of Diffie-Helman 2048-bit key exchanges on the NetScaler appliance.

sslTotRSAAuthorizations (1.3.6.1.4.1.5951.4.1.1.47.227)

Number of RSA authentications on the NetScaler appliance.

sslTotDHAauthorizations (1.3.6.1.4.1.5951.4.1.1.47.228)

Number of Diffie-Helman authentications on the NetScaler appliance.

sslTotDSSAuthorizations (1.3.6.1.4.1.5951.4.1.1.47.229)

Total number of times DSS authorization is used on the NetScaler appliance.

sslTotNULLAuthorizations (1.3.6.1.4.1.5951.4.1.1.47.230)

Number of Null authentications on the NetScaler appliance.

sslTot40BitRC4Ciphers (1.3.6.1.4.1.5951.4.1.1.47.231)

Number of RC4 40-bit cipher encryptions on the NetScaler appliance.

sslTot56BitRC4Ciphers (1.3.6.1.4.1.5951.4.1.1.47.232)

Number of RC4 56-bit cipher encryptions on the NetScaler appliance.

sslTot64BitRC4Ciphers (1.3.6.1.4.1.5951.4.1.1.47.233)

Number of RC4 64-bit cipher encryptions on the NetScaler appliance.

sslTot128BitRC4Ciphers (1.3.6.1.4.1.5951.4.1.1.47.234)

Number of RC4 128-bit cipher encryptions on the NetScaler appliance.

sslTot40BitDESCiphers (1.3.6.1.4.1.5951.4.1.1.47.235)

Number of DES 40-bit cipher encryptions on the NetScaler appliance.

sslTot56BitDESCiphers (1.3.6.1.4.1.5951.4.1.1.47.236)

Number of DES 56-bit cipher encryptions on the NetScaler appliance.

sslTot168Bit3DESCiphers (1.3.6.1.4.1.5951.4.1.1.47.237)

Number of DES 168-bit cipher encryptions on the NetScaler appliance.

sslTot40BitRC2Ciphers (1.3.6.1.4.1.5951.4.1.1.47.238)

Number of RC2 40-bit cipher encryptions on the NetScaler appliance.

sslTot56BitRC2Ciphers (1.3.6.1.4.1.5951.4.1.1.47.239)

Number of RC2 56-bit cipher encryptions on the NetScaler appliance.

sslTot128BitRC2Ciphers (1.3.6.1.4.1.5951.4.1.1.47.240)

Number of RC2 128-bit cipher encryptions on the NetScaler appliance.

sslTot128BitIDEACiphers (1.3.6.1.4.1.5951.4.1.1.47.241)

Number of IDEA 128-bit cipher encryptions on the NetScaler appliance.

sslTotNULLCiphers (1.3.6.1.4.1.5951.4.1.1.47.242)

Number of Null cipher encryptions on the NetScaler appliance.

sslTotMD5Mac (1.3.6.1.4.1.5951.4.1.1.47.243)

Number of MD5 hashes on the NetScaler appliance.

sslTotSHAMac (1.3.6.1.4.1.5951.4.1.1.47.244)

Number of SHA hashes on the NetScaler appliance.

sslTotOffloadBulkDES (1.3.6.1.4.1.5951.4.1.1.47.245)

Number of DES encryptions offloaded to the cryptography card.

sslTotOffloadRSAKeyExchanges (1.3.6.1.4.1.5951.4.1.1.47.246)

Number of RSA key exchanges offloaded to the cryptography card.

sslTotOffloadDHKeyExchanges (1.3.6.1.4.1.5951.4.1.1.47.247)

Number of DH key exchanges offloaded to the cryptography card.

sslTotOffloadSignRSA (1.3.6.1.4.1.5951.4.1.1.47.248)

Number of RSA sign operations offloaded to the cryptography card.

sslBeTotSessions (1.3.6.1.4.1.5951.4.1.1.47.260)

Number of back-end SSL sessions on the NetScaler appliance.

sslBeTotSSLv3Sessions (1.3.6.1.4.1.5951.4.1.1.47.261)

Number of back-end SSLv3 sessions on the NetScaler appliance.

sslBeTotTLSv1Sessions (1.3.6.1.4.1.5951.4.1.1.47.262)

Number of back-end TLSv1 sessions on the NetScaler appliance.

sslBeExpiredSessions (1.3.6.1.4.1.5951.4.1.1.47.263)

Number of back-end export sessions on the NetScaler appliance.

sslBeTotSessionMultiplexAttempts (1.3.6.1.4.1.5951.4.1.1.47.264)

Number of back-end SSL session multiplex attempts on the NetScaler appliance.

sslBeTotSessionMultiplexAttemptSuccess (1.3.6.1.4.1.5951.4.1.1.47.265)

Number of back-end SSL session multiplex successes on the NetScaler appliance.

sslBeTotSessionMultiplexAttemptFails (1.3.6.1.4.1.5951.4.1.1.47.266)

Number of back-end SSL session multiplex failures on the NetScaler appliance.

sslBeMaxMultiplexedSessions (1.3.6.1.4.1.5951.4.1.1.47.267)

Number of back-end SSL sessions reused on the NetScaler appliance.

sslBeTotSSLv3Handshakes (1.3.6.1.4.1.5951.4.1.1.47.268)

Number of back-end SSLv3 handshakes on the NetScaler appliance.

sslBeTotTLSv1Handshakes (1.3.6.1.4.1.5951.4.1.1.47.269)

Number of back-end TLSv1 handshakes on the NetScaler appliance.

sslBeTotSSLv3ClientAuthentications (1.3.6.1.4.1.5951.4.1.1.47.270)

Number of back-end SSLv3 client authentications on the NetScaler appliance.

sslBeTotTLSv1ClientAuthentications (1.3.6.1.4.1.5951.4.1.1.47.271)

Number of back-end TLSv1 client authentications on the NetScaler appliance.

sslBeTotRSA512keyExchanges (1.3.6.1.4.1.5951.4.1.1.47.272)

Number of back-end RSA 512-bit key exchanges on the NetScaler appliance.

sslBeTotRSA1024keyExchanges (1.3.6.1.4.1.5951.4.1.1.47.273)

Number of back-end RSA 1024-bit key exchanges on the NetScaler appliance.

sslBeTotRSA2048keyExchanges (1.3.6.1.4.1.5951.4.1.1.47.274)

Number of back-end RSA 2048-bit key exchanges on the NetScaler appliance.

sslBeTotDH512keyExchanges (1.3.6.1.4.1.5951.4.1.1.47.275)

Number of back-end DH 512-bit key exchanges on the NetScaler appliance.

sslBeTotDH1024keyExchanges (1.3.6.1.4.1.5951.4.1.1.47.276)

Number of back-end DH 1024-bit key exchanges on the NetScaler appliance.

sslBeTotDH2048keyExchanges (1.3.6.1.4.1.5951.4.1.1.47.277)

Number of back-end DH 2048-bit key exchanges on the NetScaler appliance.

sslBeTotRSAAuthorizations (1.3.6.1.4.1.5951.4.1.1.47.278)

Number of back-end RSA authentications on the NetScaler appliance.

sslBeTotDHAauthorizations (1.3.6.1.4.1.5951.4.1.1.47.279)

Number of back-end DH authentications on the NetScaler appliance.

sslBeTotDSSAuthorizations (1.3.6.1.4.1.5951.4.1.1.47.280)

Number of back-end DSS authentications on the NetScaler appliance.

sslBeTotNULLAuthorizations (1.3.6.1.4.1.5951.4.1.1.47.281)

Number of back-end null authentications on the NetScaler appliance.

sslBeTot40BitRC4Ciphers (1.3.6.1.4.1.5951.4.1.1.47.282)

Number of back-end RC4 40-bit cipher encryptions on the NetScaler appliance.

sslBeTot56BitRC4Ciphers (1.3.6.1.4.1.5951.4.1.1.47.283)

Number of back-end RC4 56-bit cipher encryptions on the NetScaler appliance.

sslBeTot64BitRC4Ciphers (1.3.6.1.4.1.5951.4.1.1.47.284)

Number of back-end RC4 64-bit cipher encryptions on the NetScaler appliance.

sslBeTot128BitRC4Ciphers (1.3.6.1.4.1.5951.4.1.1.47.285)

Number of back-end RC4 128-bit cipher encryptions on the NetScaler appliance.

sslBeTot40BitDESCiphers (1.3.6.1.4.1.5951.4.1.1.47.286)

Number of back-end DES 40-bit cipher encryptions on the NetScaler appliance.

sslBeTot56BitDESCiphers (1.3.6.1.4.1.5951.4.1.1.47.287)

Number of back-end DES 56-bit cipher encryptions on the NetScaler appliance.

sslBeTot168Bit3DESCiphers (1.3.6.1.4.1.5951.4.1.1.47.288)

Number of back-end 3DES 168-bit cipher encryptions on the NetScaler appliance.

sslBeTot40BitRC2Ciphers (1.3.6.1.4.1.5951.4.1.1.47.289)

Number of back-end RC2 40-bit cipher encryptions on the NetScaler appliance.

sslBeTot56BitRC2Ciphers (1.3.6.1.4.1.5951.4.1.1.47.290)

Number of back-end RC2 56-bit cipher encryptions on the NetScaler appliance.

sslBeTot128BitRC2Ciphers (1.3.6.1.4.1.5951.4.1.1.47.291)

Number of back-end RC2 128-bit cipher encryptions on the NetScaler appliance.

sslBeTot128BitIDEACiphers (1.3.6.1.4.1.5951.4.1.1.47.292)

Number of back-end IDEA 128-bit cipher encryptions on the NetScaler appliance.

sslBeTotNULLCiphers (1.3.6.1.4.1.5951.4.1.1.47.293)

Number of back-end null cipher encryptions on the NetScaler appliance.

sslBeTotMD5Mac (1.3.6.1.4.1.5951.4.1.1.47.294)

Number of back-end MD5 hashes on the NetScaler appliance.

sslBeTotSHAMac (1.3.6.1.4.1.5951.4.1.1.47.295)

Number of back-end SHA hashes on the NetScaler appliance.

sslCurSessions (1.3.6.1.4.1.5951.4.1.1.47.296)

Number of active SSL sessions on the NetScaler appliance.

sslTotOffloadBulkAES (1.3.6.1.4.1.5951.4.1.1.47.297)

Number of AES encryptions offloaded to the cryptography card.

sslTotOffloadBulkRC4 (1.3.6.1.4.1.5951.4.1.1.47.298)

Number of RC4 encryptions offloaded to the cryptography card.

sslNumCardsUP (1.3.6.1.4.1.5951.4.1.1.47.299)

Number of SSL cards that are UP. If the number of cards UP is lower than a threshold, a failover is initiated.

sslCards (1.3.6.1.4.1.5951.4.1.1.47.300)

Number of SSL crypto cards present on the NetScaler appliance.

sslTotBkendSessionReNegotiate (1.3.6.1.4.1.5951.4.1.1.47.301)

Number of back-end SSL session renegotiations on the NetScaler appliance.

sslTotCipherAES128 (1.3.6.1.4.1.5951.4.1.1.47.302)

Number of AES 128-bit cipher encryptions on the NetScaler appliance.

sslTotBkendSslV3Renego (1.3.6.1.4.1.5951.4.1.1.47.303)

Number of back-end SSLv3 session renegotiations on the NetScaler appliance.

sslTotBkendTlSv1Renego (1.3.6.1.4.1.5951.4.1.1.47.304)

Number of back-end TLSv1 session renegotiations on the NetScaler appliance.

sslTotCipherAES256 (1.3.6.1.4.1.5951.4.1.1.47.305)

Number of AES 256-bit cipher encryptions on the NetScaler appliance.

sslTotBkendCipherAES128 (1.3.6.1.4.1.5951.4.1.1.47.306)

Back-end AES 128-bit cipher encryptions on the NetScaler appliance.

sslTotBkendCipherAES256 (1.3.6.1.4.1.5951.4.1.1.47.307)

Back-end AES 256-bit cipher encryptions on the NetScaler appliance.

sslTotHwEncBE (1.3.6.1.4.1.5951.4.1.1.47.308)

Number of bytes encrypted in hardware on the back end.

sslTotDec (1.3.6.1.4.1.5951.4.1.1.47.309)

Number of bytes decrypted on the NetScaler appliance.

sslTotSwEncFE (1.3.6.1.4.1.5951.4.1.1.47.310)

Number of bytes encrypted in software on the front end.

sslTotEncFE (1.3.6.1.4.1.5951.4.1.1.47.311)

Number of bytes encrypted on the front end.

sslTotEnc (1.3.6.1.4.1.5951.4.1.1.47.312)

Number of bytes encrypted on the NetScaler appliance.

sslTotDecHw (1.3.6.1.4.1.5951.4.1.1.47.313)

Number of bytes decrypted in hardware.

sslTotSwDecBE (1.3.6.1.4.1.5951.4.1.1.47.314)

Number of bytes decrypted in software on back-end

sslTotHwDecFE (1.3.6.1.4.1.5951.4.1.1.47.315)

Number of bytes decrypted in hardware on the front end.

sslTotEncHw (1.3.6.1.4.1.5951.4.1.1.47.316)

Number of bytes encrypted in hardware.

sslTotDecSw (1.3.6.1.4.1.5951.4.1.1.47.317)

Number of bytes decrypted in software.

sslTotSwEncBE (1.3.6.1.4.1.5951.4.1.1.47.318)

Number of bytes encrypted in software on the back end.

sslTotEncSw (1.3.6.1.4.1.5951.4.1.1.47.319)

Number of bytes encrypted in software.

sslTotSwDecFE (1.3.6.1.4.1.5951.4.1.1.47.320)

Number of bytes decrypted in software on the front end.

sslTotEncBE (1.3.6.1.4.1.5951.4.1.1.47.321)

Number of bytes encrypted on the back end.

sslTotDecBE (1.3.6.1.4.1.5951.4.1.1.47.322)

Number of bytes decrypted on the back end.

sslTotHwDecBE (1.3.6.1.4.1.5951.4.1.1.47.323)

Number of bytes decrypted in hardware on the back end.

sslTotDecFE (1.3.6.1.4.1.5951.4.1.1.47.324)

Number of bytes decrypted on the front end.

sslTotHwEncFE (1.3.6.1.4.1.5951.4.1.1.47.325)

Number of bytes encrypted in hardware on the front end.

sslTotRSA4096keyExchanges (1.3.6.1.4.1.5951.4.1.1.47.326)

Number of RSA 4096-bit key exchanges on the NetScaler appliance.

sslCurQSize (1.3.6.1.4.1.5951.4.1.1.47.327)

Current queue size

sslChipReinitCount (1.3.6.1.4.1.5951.4.1.1.47.328)

Count of all reinitialize event for all SSL crypto chips.

sslTotECDHE224keyExchanges (1.3.6.1.4.1.5951.4.1.1.47.329)

Number of 224 Elliptical Curve Diffie-Helman on the NetScaler appliance.

sslTotECDHE256keyExchanges (1.3.6.1.4.1.5951.4.1.1.47.330)

Number of 256 Elliptical Curve Diffie-Helman on the NetScaler appliance.

sslTotECDHE384keyExchanges (1.3.6.1.4.1.5951.4.1.1.47.331)

Number of 384 Elliptical Curve Diffie-Helman on the NetScaler appliance.

sslTotECDHE521keyExchanges (1.3.6.1.4.1.5951.4.1.1.47.332)

Number of 521 Elliptical Curve Diffie-Helman on the NetScaler appliance.

sslBeTotEcdheCurve521 (1.3.6.1.4.1.5951.4.1.1.47.337)

Number of back-end ECDHE 521 curve Key exchanges on the NetScaler appliance.

sslBeTotEcdheCurve384 (1.3.6.1.4.1.5951.4.1.1.47.338)

Number of back-end ECDHE 384 curve Key exchanges on the NetScaler appliance.

sslBeTotEcdheCurve256 (1.3.6.1.4.1.5951.4.1.1.47.339)

Number of back-end ECDHE 256 curve Key exchanges on the NetScaler appliance.

sslBeTotEcdheCurve224 (1.3.6.1.4.1.5951.4.1.1.47.340)

Number of back-end ECDHE 224 curve Key exchanges on the NetScaler appliance.

sslTotTLsv11Handshakes (1.3.6.1.4.1.5951.4.1.1.47.341)

Number of SSL handshakes on TLSv1.1 on the NetScaler appliance.

sslTotTLsv12Handshakes (1.3.6.1.4.1.5951.4.1.1.47.342)

Number of SSL handshakes on TLSv1.2 on the NetScaler appliance.

sslTotTLsv11Transactions (1.3.6.1.4.1.5951.4.1.1.47.343)

Number of TLSv1.1 transactions on the NetScaler appliance.

sslTotTLsv12Transactions (1.3.6.1.4.1.5951.4.1.1.47.344)

Number of TLSv1.2 transactions on the NetScaler appliance.

sslTotTLsv11Sessions (1.3.6.1.4.1.5951.4.1.1.47.345)

Number of TLSv1.1 sessions on the NetScaler appliance.

sslTotTLsv12Sessions (1.3.6.1.4.1.5951.4.1.1.47.346)

Number of TLSv1.2 sessions on the NetScaler appliance.

sslTotTLsv11RenegSessions (1.3.6.1.4.1.5951.4.1.1.47.347)

Number of SSL session renegotiations done on TLSv1.1.

sslTotTLsv12RenegSessions (1.3.6.1.4.1.5951.4.1.1.47.348)

Number of SSL session renegotiations done on TLSv1.2.

sslTotTLsv11ClientAuthentications (1.3.6.1.4.1.5951.4.1.1.47.349)

Number of client authentications done on TLSv1.1.

sslTotTLsv12ClientAuthentications (1.3.6.1.4.1.5951.4.1.1.47.350)

Number of client authentications done on TLSv1.2.

sslTot128BitAESGCMCiphers (1.3.6.1.4.1.5951.4.1.1.47.353)

Number of AEC-GCM 128-bit cipher encryptions on the NetScaler appliance.

sslTot256BitAESGCMCiphers (1.3.6.1.4.1.5951.4.1.1.47.354)

Number of AEC-GCM 256-bit cipher encryptions on the NetScaler appliance.

sslTotOffloadBulkAESGCM128 (1.3.6.1.4.1.5951.4.1.1.47.355)

Number of AES-GCM 128-bit encryptions offloaded to the cryptography card.

sslTotOffloadBulkAESGCM256 (1.3.6.1.4.1.5951.4.1.1.47.356)

Number of AES-GCM 256-bit encryptions offloaded to the cryptography card.

sslBeTotTLsv11Sessions (1.3.6.1.4.1.5951.4.1.1.47.357)

Number of back-end TLSv1.1 sessions on the NetScaler appliance.

sslBeTotTLsv12Sessions (1.3.6.1.4.1.5951.4.1.1.47.358)

Number of back-end TLSv1.2 sessions on the NetScaler appliance.

sslBeTotTLsv11Handshakes (1.3.6.1.4.1.5951.4.1.1.47.359)

Number of back-end TLSv1.1 handshakes on the NetScaler appliance.

sslBeTotTLsV12Handshakes (1.3.6.1.4.1.5951.4.1.1.47.360)

Number of back-end TLSv1.2 handshakes on the NetScaler appliance.

sslBeTotTLsV11ClientAuthentications (1.3.6.1.4.1.5951.4.1.1.47.361)

Number of back-end TLSv1.1 client authentications on the NetScaler appliance.

sslBeTotTLsV12ClientAuthentications (1.3.6.1.4.1.5951.4.1.1.47.362)

Number of back-end TLSv1.2 client authentications on the NetScaler appliance.

sslTotBkendTISv11Renego (1.3.6.1.4.1.5951.4.1.1.47.363)

Number of back-end TLSv1.1 session renegotiations on the NetScaler appliance.

sslTotBkendTISv12Renego (1.3.6.1.4.1.5951.4.1.1.47.364)

Number of back-end TLSv1.2 session renegotiations on the NetScaler appliance.

nsHttpStatsGroup (1.3.6.1.4.1.5951.4.1.1.48)**httpTotGets (1.3.6.1.4.1.5951.4.1.1.48.45)**

Total number of HTTP requests received with the GET method.

httpTotPosts (1.3.6.1.4.1.5951.4.1.1.48.46)

Total number of HTTP requests received with the POST method.

httpTotOthers (1.3.6.1.4.1.5951.4.1.1.48.47)

Total number of HTTP requests received with methods other than GET and POST. Some of the other well-defined HTTP methods are HEAD, PUT, DELETE, OPTIONS, and TRACE. User-defined methods are also allowed.

httpTotRxRequestBytes (1.3.6.1.4.1.5951.4.1.1.48.48)

Total number of bytes of HTTP request data received.

httpTotRxResponseBytes (1.3.6.1.4.1.5951.4.1.1.48.49)

Total number of bytes of HTTP response data received.

httpTotTxRequestBytes (1.3.6.1.4.1.5951.4.1.1.48.50)

Total number of bytes of HTTP request data transmitted.

httpTotTxResponseBytes (1.3.6.1.4.1.5951.4.1.1.48.51)

Total number of bytes of HTTP response data transmitted.

httpTot10Requests (1.3.6.1.4.1.5951.4.1.1.48.52)

Total number of HTTP/1.0 requests received.

httpTotResponses (1.3.6.1.4.1.5951.4.1.1.48.53)

Total number of HTTP responses sent.

httpTot10Responses (1.3.6.1.4.1.5951.4.1.1.48.54)

Total number of HTTP/1.0 responses sent.

httpTotClenResponses (1.3.6.1.4.1.5951.4.1.1.48.55)

Total number of HTTP responses sent in which the Content-length field of the HTTP header has been set. Content-length specifies the length of the content, in bytes, in the associated HTTP body.

httpTotChunkedResponses (1.3.6.1.4.1.5951.4.1.1.48.56)

Total number of HTTP responses sent in which the Transfer-Encoding field of the HTTP header has been set to chunked. This setting is used when the server wants to start sending the response before knowing its total length. The server breaks the response into chunks and sends them in sequence, inserting the length of each chunk before the actual data. The message ends with a chunk of size zero.

httpErrIncompleteRequests (1.3.6.1.4.1.5951.4.1.1.48.57)

Total number of HTTP requests received in which the header spans more than one packet.

httpErrIncompleteResponses (1.3.6.1.4.1.5951.4.1.1.48.58)

Total number of HTTP responses received in which the header spans more than one packet.

httpErrIncompleteHeaders (1.3.6.1.4.1.5951.4.1.1.48.60)

Total number of HTTP requests and responses received in which the HTTP header spans more than one packet.

httpErrServerBusy (1.3.6.1.4.1.5951.4.1.1.48.61)

Total number of HTTP error responses received. Some of the error responses are:

500 Internal Server Error

501 Not Implemented

502 Bad Gateway

503 Service Unavailable

504 Gateway Timeout

505 HTTP Version Not Supported

httpTotChunkedRequests (1.3.6.1.4.1.5951.4.1.1.48.62)

Total number of HTTP requests in which the Transfer-Encoding field of the HTTP header has been set to chunked.

httpTotClenRequests (1.3.6.1.4.1.5951.4.1.1.48.63)

Total number of HTTP requests in which the Content-length field of the HTTP header has been set. Content-length specifies the length of the content, in bytes, in the associated HTTP body.

httpErrLargeContent (1.3.6.1.4.1.5951.4.1.1.48.64)

Total number of requests and responses received with large body.

httpErrLargeCtlen (1.3.6.1.4.1.5951.4.1.1.48.65)

Total number of requests received with large content, in which the Content-length field of the HTTP header has been set. Content-length specifies the length of the content, in bytes, in the associated HTTP body.

httpErrLargeChunk (1.3.6.1.4.1.5951.4.1.1.48.66)

Total number of requests received with large chunk size, in which the Transfer-Encoding field of the HTTP header has been set to chunked.

httpTotRequests (1.3.6.1.4.1.5951.4.1.1.48.67)

Total number of HTTP requests received.

httpTot11Requests (1.3.6.1.4.1.5951.4.1.1.48.68)

Total number of HTTP/1.1 requests received.

httpTot11Responses (1.3.6.1.4.1.5951.4.1.1.48.69)

Total number of HTTP/1.1 responses sent.

httpTotNoClenChunkResponses (1.3.6.1.4.1.5951.4.1.1.48.70)

Total number of FIN-terminated responses sent. In FIN-terminated responses, the server finishes sending the data and closes the connection.

httpErrNoreuseMultipart (1.3.6.1.4.1.5951.4.1.1.48.71)

Total number of HTTP multi-part responses sent. In multi-part responses, one or more entities are encapsulated within the body of a single message.

spdyTotStreams (1.3.6.1.4.1.5951.4.1.1.48.73)

Total number of requests received over SPDYv2 and SPDYv3

spdyv2TotStreams (1.3.6.1.4.1.5951.4.1.1.48.74)

Total number of requests received over SPDYv2

spdyv3TotStreams (1.3.6.1.4.1.5951.4.1.1.48.75)

Total number of requests received over SPDYv3

nsCacheStatsGroup (1.3.6.1.4.1.5951.4.1.1.49)

cacheMaxMemoryKB (1.3.6.1.4.1.5951.4.1.1.49.1)

Largest amount of memory the NetScaler can dedicate to caching, up to 50% of available memory. A 0 value disables caching, but the caching module continues to run.

cacheUtilizedMemoryKB (1.3.6.1.4.1.5951.4.1.1.49.2)

Amount of memory the integrated cache is currently using.

cacheNumCached (1.3.6.1.4.1.5951.4.1.1.49.3)

Responses currently in integrated cache. Includes responses fully downloaded, in the process of being downloaded, and expired or flushed but not yet removed.

cachePercentHit (1.3.6.1.4.1.5951.4.1.1.49.12)

Cache hits as percentage of the total number of requests

cacheRecentPercentHit (1.3.6.1.4.1.5951.4.1.1.49.13)

Recently recorded cache hit ratio expressed as percentage

cacheCurHits (1.3.6.1.4.1.5951.4.1.1.49.14)

This number should be close to the number of hits being served currently.

cacheCurMisses (1.3.6.1.4.1.5951.4.1.1.49.15)

Responses fetched from the origin and served from the cache. Should approximate storable misses. Does not include non-storable misses.

cachePercent304Hits (1.3.6.1.4.1.5951.4.1.1.49.20)

304 responses as a percentage of all responses that the NetScaler served.

cacheRecentPercent304Hits (1.3.6.1.4.1.5951.4.1.1.49.21)

Recently recorded ratio of 304 hits to all hits expressed as percentage

cachePercentStoreAbleMiss (1.3.6.1.4.1.5951.4.1.1.49.26)

Responses that were fetched from the origin, stored in the cache, and then served to the client, as a percentage of all cache misses.

cacheRecentPercentStoreAbleMiss (1.3.6.1.4.1.5951.4.1.1.49.27)

Recently recorded ratio of store-able misses to all misses expressed as percentage.

cachePercentSuccessfulRevalidation (1.3.6.1.4.1.5951.4.1.1.49.34)

Percentage of times stored content was successfully revalidated by a 304 (Object Not Modified) response rather than by a full response

cacheRecentPercentSuccessfulRevalidation (1.3.6.1.4.1.5951.4.1.1.49.35)

Recently recorded percentage of times stored content was successfully revalidated by a 304 response rather than by a full response

cachePercentByteHit (1.3.6.1.4.1.5951.4.1.1.49.40)

Bytes served from the cache divided by total bytes served to the client. If compression is On in the NetScaler, this ratio may not reflect the bytes served by the compression module. If the compression is Off, this ratio is the same as cachePercentOriginBandwidthSaved.

cacheRecentPercentByteHit (1.3.6.1.4.1.5951.4.1.1.49.41)

Recently recorded cache byte hit ratio expressed as percentage. Here we define byte hit ratio as ((number of bytes served from the cache)/(total number of bytes served to the client)). This is the standard definition of Byte Hit Ratio. If compression is turned ON in NS then this ratio doesn't mean much. This might under or over estimate the origin-to-cache bandwidth saving (depending upon whether bytes served by CMP in NetScaler are more or less than compressed bytes served from the cache). If CMP is turned OFF in NS then this ratio is same as cacheRecentPercentOriginBandwidthSaved.

cachePercentOriginBandwidthSaved (1.3.6.1.4.1.5951.4.1.1.49.42)

Percentage of origin bandwidth saved, expressed as number of bytes served from the integrated cache divided by all bytes served. The assumption is that all compression is done in the NetScaler.

cacheRecentPercentOriginBandwidthSaved (1.3.6.1.4.1.5951.4.1.1.49.43)

Bytes served from cache divided by total bytes served to client. This ratio can be greater than 1 because of the assumption that all compression has been done in the NetScaler.

cacheErrMemAlloc (1.3.6.1.4.1.5951.4.1.1.49.44)

Total number of times the cache failed to allocate memory to store responses.

cacheTotRequests (1.3.6.1.4.1.5951.4.1.1.49.45)

Total cache hits plus total cache misses.

cacheTotHits (1.3.6.1.4.1.5951.4.1.1.49.46)

Responses served from the integrated cache. These responses match a policy with a CACHE action.

cacheTotMisses (1.3.6.1.4.1.5951.4.1.1.49.47)

Intercepted HTTP requests requiring fetches from origin server.

cacheTot304Hits (1.3.6.1.4.1.5951.4.1.1.49.48)

Object not modified responses served from the cache. (Status code 304 served instead of the full response.)

cacheTotNon304Hits (1.3.6.1.4.1.5951.4.1.1.49.49)

Total number of full (non-304) responses served from the cache. A 304 status code indicates that a response has not been modified since the last time it was served

cacheTotStoreAbleMisses (1.3.6.1.4.1.5951.4.1.1.49.50)

Cache misses for which the fetched response is stored in the cache before serving it to the client. Storable misses conform to a built-in or user-defined caching policy that contains a CACHE action.

cacheTotNonStoreAbleMisses (1.3.6.1.4.1.5951.4.1.1.49.51)

Cache misses for which the fetched response is not stored in the cache. These responses match policies with a NOCACHE action or are affected by Poll Every Time.

cacheTotRevalidationMiss (1.3.6.1.4.1.5951.4.1.1.49.52)

Responses that an intervening cache revalidated with the integrated cache before serving, as determined by a Cache-Control: Max-Age header configurable in the integrated cache

cacheTotFullToConditionalRequest (1.3.6.1.4.1.5951.4.1.1.49.53)

Number of user-agent requests for a cached Poll Every Time (PET) response that were sent to the origin server as conditional requests.

cacheTotSuccessfulRevalidation (1.3.6.1.4.1.5951.4.1.1.49.54)

Total number of times stored content was successfully revalidated by a 304 Not Modified response from the origin.

cacheTotResponseBytes (1.3.6.1.4.1.5951.4.1.1.49.55)

Total number of HTTP response bytes served by NetScaler from both the origin and the cache

cacheBytesServed (1.3.6.1.4.1.5951.4.1.1.49.56)

Total number of bytes served from the integrated cache

cacheCompressedBytesServed (1.3.6.1.4.1.5951.4.1.1.49.57)

Number of compressed bytes served from the cache

cacheTotPetRequests (1.3.6.1.4.1.5951.4.1.1.49.58)

Requests that triggered a search of a content group that has Poll Every Time (PET) enabled (always consult the origin server before serving cached data).

cacheTotPetHits (1.3.6.1.4.1.5951.4.1.1.49.59)

Number of times a cache hit was found during a search of a content group that has Poll Every Time enabled.

cachePercentPetHits (1.3.6.1.4.1.5951.4.1.1.49.60)

Percentage of cache hits in content groups that have Poll Every Time enabled, relative to all searches of content groups with Poll Every Time enabled.

cacheTotParameterizedRequests (1.3.6.1.4.1.5951.4.1.1.49.61)

Total number of requests where the content group has hit and invalidation parameters or selectors.

cacheTotParameterizedHits (1.3.6.1.4.1.5951.4.1.1.49.62)

Parameterized requests resulting in either a 304 or non-304 hit.

cacheTotParameterizedNon304Hits (1.3.6.1.4.1.5951.4.1.1.49.63)

Parameterized requests resulting in a full response (not status code 304: Object Not Updated) served from the cache.

cacheTotParameterized304Hits (1.3.6.1.4.1.5951.4.1.1.49.64)

Parameterized requests resulting in an object not modified (status code 304) response.

cachePercentParameterized304Hits (1.3.6.1.4.1.5951.4.1.1.49.65)

Percentage of parameterized 304 hits relative to all parameterized hits.

cacheRecentPercentParameterizedHits (1.3.6.1.4.1.5951.4.1.1.49.66)

Recently recorded ratio of parameterized 304 hits to all parameterized hits expressed as a percentage

cacheTotInvalidationRequests (1.3.6.1.4.1.5951.4.1.1.49.67)

Requests that match an invalidation policy and result in expiration of specific cached responses or entire content groups.

cacheTotNonParameterizedInvalidationRequests (1.3.6.1.4.1.5951.4.1.1.49.68)

Requests that match an invalidation policy where the invalGroups parameter is configured and expires one or more content groups.

cacheTotParameterizedInvalidationRequests (1.3.6.1.4.1.5951.4.1.1.49.69)

Requests matching a policy with an invalidation (INVAL) action and a content group that uses an invalidation selector or parameters.

cacheLargestResponseReceived (1.3.6.1.4.1.5951.4.1.1.49.70)

Size, in bytes, of largest response sent to client from the cache or the origin server.

cacheTotFlashcacheMisses (1.3.6.1.4.1.5951.4.1.1.49.71)

Number of requests to a content group with flash cache enabled that were cache misses. Flash cache distributes the response to all the clients in a queue.

cacheTotFlashcacheHits (1.3.6.1.4.1.5951.4.1.1.49.72)

Number of requests to a content group with flash cache enabled that were cache hits. The flash cache setting queues requests that arrive simultaneously and distributes the response to all the clients in the queue.

cacheTotExpireAtLastByte (1.3.6.1.4.1.5951.4.1.1.49.73)

Instances of content expiring immediately after receiving the last body byte due to the Expire at Last Byte setting for the content group.

cacheNumMarker (1.3.6.1.4.1.5951.4.1.1.49.74)

Marker objects created when a response exceeds the maximum or minimum size for entries in its content group or has not yet received the minimum number of hits required for items in its content group.

cacheMaxMemoryActiveKB (1.3.6.1.4.1.5951.4.1.1.49.75)

Currently active value of maximum memory

cache64MaxMemoryKB (1.3.6.1.4.1.5951.4.1.1.49.76)

Largest amount of memory the NetScaler can dedicate to caching, up to 50% of available memory. A 0 value disables caching, but the caching module continues to run.

cacheNumObjSavedOnDisk (1.3.6.1.4.1.5951.4.1.1.49.77)

Cached responses currently saved on disk. Includes responses fully saved to disk, and expired or flushed but not yet removed.

cacheNumMBSavedOnDisk (1.3.6.1.4.1.5951.4.1.1.49.78)

Size (MB) of cached responses currently saved on disk. Includes responses fully saved to disk, and expired or flushed but not yet removed.

cacheNumMBReadFromDisk (1.3.6.1.4.1.5951.4.1.1.49.79)

Total Number of MB read from disk since last reboot.

cacheNumMBWrittenToDisk (1.3.6.1.4.1.5951.4.1.1.49.80)

Total Number of MB written to disk since last reboot.

cacheTotSqlHits (1.3.6.1.4.1.5951.4.1.1.49.81)

sql response served from cache

nsCompressionStatsGroup (1.3.6.1.4.1.5951.4.1.1.50)

compTotalRequests (1.3.6.1.4.1.5951.4.1.1.50.1)

Number of HTTP compression requests the NetScaler receives for which the response is successfully compressed. For example, after you enable compression and configure services, if you send requests to the NetScaler with the following header information: ?Accept-Encoding: gzip, deflate?, and NetScaler compresses the corresponding response, this counter is incremented.

compTotalTxBytes (1.3.6.1.4.1.5951.4.1.1.50.2)

Number of bytes the NetScaler sends to the client after compressing the response from the server.

compTotalRxBytes (1.3.6.1.4.1.5951.4.1.1.50.3)

Number of bytes that can be compressed, which the NetScaler receives from the server. This gives the content length of the response that the NetScaler receives from server.

compTotalTxPackets (1.3.6.1.4.1.5951.4.1.1.50.4)

Number of HTTP packets that the NetScaler sends to the client after compressing the response from the server.

compTotalRxPackets (1.3.6.1.4.1.5951.4.1.1.50.5)

Number of HTTP packets that can be compressed, which the NetScaler receives from the server.

compRatio (1.3.6.1.4.1.5951.4.1.1.50.6)

Ratio of compressible data received to compressed data transmitted expressed as percentage.

compTotalDataCompressionRatio (1.3.6.1.4.1.5951.4.1.1.50.7)

Ratio of total HTTP data received to total HTTP data transmitted expressed as percentage.

compTcpTotalTxBytes (1.3.6.1.4.1.5951.4.1.1.50.8)

Number of bytes that the NetScaler sends to the client after compressing the response from the server.

compTcpTotalRxBytes (1.3.6.1.4.1.5951.4.1.1.50.9)

Number of bytes that can be compressed, which the NetScaler receives from the server. This gives the content length of the response that the NetScaler receives from server.

compTcpTotalTxPackets (1.3.6.1.4.1.5951.4.1.1.50.10)

Number of TCP packets that the NetScaler sends to the client after compressing the response from the server.

compTcpTotalRxPackets (1.3.6.1.4.1.5951.4.1.1.50.11)

Total number of compressible packets received by NetScaler.

compTcpTotalQuantum (1.3.6.1.4.1.5951.4.1.1.50.12)

Number of times the NetScaler compresses a quantum of data. NetScaler buffers the data received from the server till it reaches the quantum size and then compresses the buffered data and transmits to the client.

compTcpTotalPush (1.3.6.1.4.1.5951.4.1.1.50.13)

Number of times the NetScaler compresses data on receiving a TCP PUSH flag from the server. The PUSH flag ensures that data is compressed immediately without waiting for the buffered data size to reach the quantum size.

compTcpTotalEoi (1.3.6.1.4.1.5951.4.1.1.50.14)

Number of times the NetScaler compresses data on receiving End Of Input (FIN packet). When the NetScaler receives End Of Input (FIN packet), it compresses the buffered data immediately without waiting for the buffered data size to reach the quantum size.

compTcpTotalTimer (1.3.6.1.4.1.5951.4.1.1.50.15)

Number of times the NetScaler compresses data on expiration of data accumulation timer. The timer expires if the server response is very slow and consequently, the NetScaler does not receive response for a certain amount of time. Under such a condition, the NetScaler compresses the buffered data immediately without waiting for the buffered data size to reach the quantum size.

compTcpRatio (1.3.6.1.4.1.5951.4.1.1.50.16)

Ratio of compressible data received to compressed data transmitted expressed as percentage.

compTcpBandwidthSaving (1.3.6.1.4.1.5951.4.1.1.50.17)

Bandwidth saving from TCP compression expressed as percentage.

deCompTcpRxPackets (1.3.6.1.4.1.5951.4.1.1.50.18)

Total number of compressed packets received by NetScaler.

deCompTcpTxPackets (1.3.6.1.4.1.5951.4.1.1.50.19)

Total number of decompressed packets transmitted by NetScaler.

deCompTcpRxBytes (1.3.6.1.4.1.5951.4.1.1.50.20)

Total number of compressed bytes received by NetScaler.

deCompTcpTxBytes (1.3.6.1.4.1.5951.4.1.1.50.21)

Total number of decompressed bytes transmitted by NetScaler.

deCompTcpErrData (1.3.6.1.4.1.5951.4.1.1.50.22)

Number of data errors encountered while decompressing.

deCompTcpErrLessData (1.3.6.1.4.1.5951.4.1.1.50.23)

Number of times NetScaler received less data than declared by protocol.

deCompTcpErrMoreData (1.3.6.1.4.1.5951.4.1.1.50.24)

Number of times NetScaler received more data than declared by protocol.

deCompTcpErrMemory (1.3.6.1.4.1.5951.4.1.1.50.25)

Number of times memory failures occurred while decompressing.

deCompTcpErrUnknown (1.3.6.1.4.1.5951.4.1.1.50.26)

Number of times unknown errors occurred while decompressing.

deCompTcpRatio (1.3.6.1.4.1.5951.4.1.1.50.27)

Ratio of decompressed data transmitted to compressed data received expressed as percentage.

deCompTcpBandwidthSaving (1.3.6.1.4.1.5951.4.1.1.50.28)

Bandwidth saving from compression expressed as percentage.

delCompTotalRequests (1.3.6.1.4.1.5951.4.1.1.50.29)

Total number of delta compression requests received by NetScaler.

delCompFirstAccess (1.3.6.1.4.1.5951.4.1.1.50.30)

Total number of delta compression first accesses.

delCompDone (1.3.6.1.4.1.5951.4.1.1.50.31)

Total number of delta compressions done by NetScaler.

delCompTcpRxBytes (1.3.6.1.4.1.5951.4.1.1.50.32)

Total number of delta-compressible bytes received by NetScaler.

delCompTcpTxBytes (1.3.6.1.4.1.5951.4.1.1.50.33)

Total number of delta-compressed bytes transmitted by NetScaler.

delCompTcpRxPackets (1.3.6.1.4.1.5951.4.1.1.50.34)

Number of delta-compressible packets received.

delCompTcpTxPackets (1.3.6.1.4.1.5951.4.1.1.50.35)

Total number of delta-compressed packets transmitted by NetScaler.

delCompBaseServed (1.3.6.1.4.1.5951.4.1.1.50.36)

Total number of basefile requests served by NetScaler.

delCompBaseTcpTxBytes (1.3.6.1.4.1.5951.4.1.1.50.37)

Number of basefile bytes transmitted by NetScaler.

delCompErrBypassed (1.3.6.1.4.1.5951.4.1.1.50.39)

Number of times delta-compression bypassed by NetScaler.

delCompErrBFileWHdrFailed (1.3.6.1.4.1.5951.4.1.1.50.40)

Number of times basefile could not be updated in NetScaler cache.

delCompErrNostoreMiss (1.3.6.1.4.1.5951.4.1.1.50.41)

Number of times basefile was not found in NetScaler cache.

delCompErrReqinfoToobig (1.3.6.1.4.1.5951.4.1.1.50.42)

Number of times basefile request URL was too large.

delCompErrReqinfoAllocfail (1.3.6.1.4.1.5951.4.1.1.50.43)

Number of times requested basefile could not be allocated.

delCompErrSessallocFail (1.3.6.1.4.1.5951.4.1.1.50.44)

Number of times delta compression session could not be allocated.

delCmpRatio (1.3.6.1.4.1.5951.4.1.1.50.45)

Ratio of compressible data received to compressed data transmitted expressed as percentage.

delBwSaving (1.3.6.1.4.1.5951.4.1.1.50.46)

Bandwidth saving from delta compression expressed as percentage.

compHttpBandwidthSaving (1.3.6.1.4.1.5951.4.1.1.50.47)

Bandwidth saving from TCP compression expressed as percentage.

nsIfStatsTable (1.3.6.1.4.1.5951.4.1.1.54)

The interface related statistics Table.

Indexed on: [ifName](#)

ifName (1.3.6.1.4.1.5951.4.1.1.54.1.1)

The name of the interface.

ifMedia (1.3.6.1.4.1.5951.4.1.1.54.1.2)

The media type of the interface.

ifTotRxBytes (1.3.6.1.4.1.5951.4.1.1.54.1.3)

Number of bytes received by an interface since the NetScaler appliance was started or the interface statistics were cleared.

ifRxAvgBandwidthUsage (1.3.6.1.4.1.5951.4.1.1.54.1.4)

The average bandwidth, in bits per second, at which the specified interface has been receiving packets since the NetScaler appliance was started or the interface statistics were cleared.

ifTotRxPkts (1.3.6.1.4.1.5951.4.1.1.54.1.5)

Number of packets received by an interface since the NetScaler appliance was started or the interface statistics were cleared.

ifRxAvgPacketRate (1.3.6.1.4.1.5951.4.1.1.54.1.6)

Average rate, in packets per second, of incoming packets on the specified interface since the NetScaler appliance was started or the interface statistics were cleared.

ifTotTxBytes (1.3.6.1.4.1.5951.4.1.1.54.1.7)

Number of bytes transmitted by an interface since the NetScaler appliance was started or the interface statistics were cleared.

ifTxAvgBandwidthUsage (1.3.6.1.4.1.5951.4.1.1.54.1.8)

The average bandwidth, in bits per second, at which the specified interface has been transmitting packets since the NetScaler appliance was started or the interface statistics were cleared.

ifTotTxPkts (1.3.6.1.4.1.5951.4.1.1.54.1.9)

Number of packets transmitted by an interface since the NetScaler appliance was started or the interface statistics were cleared.

ifTxAvgPacketRate (1.3.6.1.4.1.5951.4.1.1.54.1.10)

Average rate, in packets per second, of outgoing packets on the specified interface since the NetScaler appliance was started or the interface statistics were cleared.

ifRxCRCErrors (1.3.6.1.4.1.5951.4.1.1.54.1.11)

Number of packets received with the wrong checksum by the specified interface since the NetScaler appliance was started or the interface statistics were cleared. This indicates the number of Jabber frames received instead of CRC errors on the 10G data ports of NetScaler 12000-10G platform and the data ports of NetScaler MPX 15000 and 17000 platforms.

ifRxFrameErrors (1.3.6.1.4.1.5951.4.1.1.54.1.12)

Number of Jumbo frames(frame size greater than 1518 bytes) received by the specified interface since the NetScaler appliance was started or the interface statistics were cleared.

ifRxAlignmentErrors (1.3.6.1.4.1.5951.4.1.1.54.1.13)

Number of packets received with an alignment error (an error that occurs when the total number of bits of a received frame is not divisible by eight) by the specified interface. Since the NetScaler appliance was started or the interface statistics were cleared.

ifTxCollisions (1.3.6.1.4.1.5951.4.1.1.54.1.14)

Number of collisions detected during transmission, on the specified interface, since the NetScaler appliance was started or the interface statistics were cleared. This statistic is applicable only to half-duplex transmissions and is available only on the data ports of the NetScaler 12000 platform and the management port of the NetScaler MPX 15000 and 17000 platforms.

ifTxExcessCollisions (1.3.6.1.4.1.5951.4.1.1.54.1.15)

Number of excess collisions detected during transmission, on the specified interface, since the NetScaler appliance was started or the interface statistics were cleared. This statistic is only applicable to half-duplex transmissions and is supported only on the NetScaler Classic edition.

ifTxLateCollisions (1.3.6.1.4.1.5951.4.1.1.54.1.16)

Number of late collisions detected during transmission, on the specified interface, since the NetScaler appliance was started or the interface statistics were cleared. This statistic is only applicable to half-duplex transmissions. Currently this is supported only on Classic builds.

ifTxMultiCollisionErrors (1.3.6.1.4.1.5951.4.1.1.54.1.17)

Number of multiple collisions during transmission on the specified interface. This counter is deprecated now. (Includes only half-duplex transmissions.)

ifTxCarrierError (1.3.6.1.4.1.5951.4.1.1.54.1.18)

Number of Carrier Sense errors that occur when an interface attempts to transmit a frame but is unable to do so as no carrier is detected. This statistic is applicable only to half-duplex transmissions and is available only on 1G data ports of the NetScaler 12000 platform and management ports of NetScaler MPX 15000 and 17000 platforms.

(1) Loop back interface (LO) of all platforms indicates PE fails to send packets to BSD stack because of lack of resources in BSD stack.

(2) Number of non Cisco Heart beat packet drop on internal interface. Applicable for internal interface(which is used to communicate between NXOS-NSVSB) of VPX's running on Cisco Nexus platform

ifTotRxMbits (1.3.6.1.4.1.5951.4.1.1.54.1.19)

The total data, in megabits, received by an interface since the NetScaler appliance was started or the interface statistics were cleared. This statistic also includes the Ethernet overhead bytes, i.e. preamble, inter-packet gap, and CRC.

ifTotTxMbits (1.3.6.1.4.1.5951.4.1.1.54.1.20)

The total data, in megabits, transmitted by an interface since the NetScaler appliance was started or the interface statistics were cleared. This statistic also includes the Ethernet overhead bytes, i.e. preamble, inter-packet gap, and CRC.

ifTotNetScalerPkts (1.3.6.1.4.1.5951.4.1.1.54.1.21)

Number of packets, destined to the NetScaler, received by an interface since the NetScaler appliance was started or the interface statistics were cleared. The packets destined to NetScaler are those that have the same MAC address as that of an interface or a VMAC address owned by the NetScaler.

ifErrDroppedRxPkts (1.3.6.1.4.1.5951.4.1.1.54.1.22)

Number of inbound packets dropped by the specified interface. Commonly dropped packets are multicast frames, spanning tree BPDUs, packets destined to a MAC not owned by the NetScaler when L2 mode is disabled, or packets tagged for a VLAN that is not bound to the interface. This statistic will increment in most healthy networks at a steady rate regardless of traffic load. If a sharp spike in dropped packets occurs, it generally indicates an issue with connected L2 switches, such as a forwarding database overflow resulting in packets being broadcast on all ports.

ifErrLinkHangs (1.3.6.1.4.1.5951.4.1.1.54.1.23)

Number of times the specified interface detected hangs in the transmit and receive paths since the NetScaler appliance was started or the interface statistics were cleared.

ifLinkReinits (1.3.6.1.4.1.5951.4.1.1.54.1.24)

Number of times the link has been re-initialized. A re-initialization occurs due to link state change, configuration parameter change, or administrative reset operation.

ifErrDuplexMismatch (1.3.6.1.4.1.5951.4.1.1.54.1.25)

Number of times duplex mismatches were detected on the specified interface since the NetScaler appliance was started or the interface statistics were cleared. A mismatch will occur if the duplex mode is not identically set on both ends of the link. This statistic is only available on the NetScaler Classic edition.

ifErrCongestedPktsDrops (1.3.6.1.4.1.5951.4.1.1.54.1.26)

Number of outbound packets dropped from the normal and low-priority transmit (Tx) overflow queues, during congestion on the specified interface, since the NetScaler appliance was started or the interface statistics were cleared. This could be caused by one of the following:

- 1) Packets that have been in the overflow queue for more than 10 milliseconds.
- 2) Shortage of free receive buffers on the NetScaler.

ifErrCongestionLimitPktDrops (1.3.6.1.4.1.5951.4.1.1.54.1.27)

Number of transmit (Tx) packets dropped from normal and low-priority transmit overflow queues, during congestion on the specified interface, since the NetScaler appliance was started or the interface statistics were cleared. This is caused when the overflow queue limits are exceeded.

ifErrPktRx (1.3.6.1.4.1.5951.4.1.1.54.1.28)

Number of inbound packets dropped by the hardware on a specified interface once the NetScaler appliance starts or the interface statistics are cleared. This happens due to following reasons:

- 1) The hardware receives packets at a rate higher rate than that at which the software is processing packets. In this case, the hardware FIFO overruns and starts dropping the packets .
- 2) The specified interface fails to receive inbound packets from the appliance because of insufficient memory.
- 3) The specified interface receives packets with CRC errors (Alignment or Frame Check Sequence).
- 4) The specified interface receives overly long packets.
- 5) The specified interface receives packets with alignment errors.
- 6) The software does less buffering because it is running out of available memory. When hardware detects that there is no space into which to push newly arrived packets, it starts dropping them.
- 7) The specified interface receives packets with Frame Check Sequence (FCS) errors.
- 8) The specified interface receives packets smaller than 64 bytes.
- 9) The specified interface discards error-free inbound packets because of insufficient resources. For example: NIC buffers.
- 10) Packets are missed because of collision detection, link lost, physical decoding error, or MAC abort.

ifErrRxFIFO (1.3.6.1.4.1.5951.4.1.1.54.1.29)

Number packet drops due to insufficient space in the receive queue, on the specified interface, since the NetScaler appliance was started or the interface statistics were cleared. This statistic is only available on the data ports of the NetScaler 12000 platform and the NetScaler MPX 15000 and 17000 platforms.

ifErrRxNoBufs (1.3.6.1.4.1.5951.4.1.1.54.1.30)

Number of times the NIC hardware reported an error, due to packets drops caused by lack of buffers. This statistic is available on:

- (1) All ports except management ports on NetScaler MPX 15000 and 17000 platforms.
- (2) All 1G ports on NetScaler MPX 7500, 9500, 10500, 12500, 15500, 17500, 19500, and 21500 platforms.

ifErrRxFCS (1.3.6.1.4.1.5951.4.1.1.54.1.32)

Number of packets received with Frame Check Sequence(FCS) errors, on the specified interface, since the NetScaler appliance was started or the interface statistics were cleared. This statistic is available only on data ports of the NetScaler 12000 platform and NetScaler MPX 15000 and 17000 platforms.

ifErrPktTx (1.3.6.1.4.1.5951.4.1.1.54.1.33)

Number of outbound packets dropped by the hardware on a specified interface since the NetScaler appliance was started or the interface statistics were cleared. This could happen due to length (undersize or oversize) errors and lack of resources. This statistic is available only for:

- (1) Loop back interface (LO) of all platforms.
- (2) All data ports on the NetScaler 12000 platform.
- (3) Management ports on the MPX 15000 and 17000 platforms.

ifErrTxFIFO (1.3.6.1.4.1.5951.4.1.1.54.1.34)

Number of times the hardware reported an error in accumulating packets for transmission on the specified interface. This statistic is only available on 10G ports of the NetScaler 12000-10G platform and data ports of the NetScaler MPX 15000 and 17000 platforms.

(1) Loop back interfaces (LO) of all platform indicates packets crossed allowed limit, which is 10K PPS on LO interface

ifErrTxHeartBeat (1.3.6.1.4.1.5951.4.1.1.54.1.35)

Number of 10Mb link heartbeats on the specified interface. This counter is deprecated now. (Informational - 10MbMb half-duplex only)

ifErrTxOverflow (1.3.6.1.4.1.5951.4.1.1.54.1.36)

Number of packets that have passed through the overflow queues, during transmission on the specified interface, since the NetScaler appliance was started or the interface statistics were cleared. This gets incremented only on congested ports.

ifErrTxDeferred (1.3.6.1.4.1.5951.4.1.1.54.1.37)

Number of times packet transmission was deferred on the specified interface. This statistic is only available on data ports of the NetScaler 12000 platform and management ports of the NetScaler MPX 15000 and 17000 platforms. This statistic is only applicable to half-duplex transmissions.

ifErrDroppedTxPkts (1.3.6.1.4.1.5951.4.1.1.54.1.38)

Number of packets dropped in transmission by the specified interface due to one of the following reasons.

- (1) VLAN mismatch.
- (2) Oversized packets.
- (3) Interface congestion.
- (4) Loopback packets sent on non loop back interface.

ifTotRxXonPause (1.3.6.1.4.1.5951.4.1.1.54.1.39)

Number of Pause frames received on the specified interface with pause time set to zero. This statistic is not available on 10G ports of the NetScaler 12000-10G platform and data ports of the NetScaler MPX 15000 and 17000 platforms.

ifTotRxXoffPause (1.3.6.1.4.1.5951.4.1.1.54.1.40)

Number of Pause frames received by the specified interface with the pause time greater than zero. This statistic is only available on 10G ports of the NetScaler 12000-10G platform and data ports of the NetScaler MPX 15000 and 17000 platforms.

ifTotXoffStateEntered (1.3.6.1.4.1.5951.4.1.1.54.1.41)

Number of times transmission was stopped on the specified interface due to the receipt of Pause frames. This statistic is only available on 10G ports of the NetScaler 12000-10G platform and data ports of the NetScaler MPX 15000 and 17000 platforms.

ifTotXonSent (1.3.6.1.4.1.5951.4.1.1.54.1.42)

Number of pause frames, sent by the specified interface, with pause time set to zero to restart transmission. This statistic is not available on 10G ports of the NetScaler 12000-10G platform and data ports of the NetScaler MPX 15000 and 17000 platforms.

ifTotXoffSent (1.3.6.1.4.1.5951.4.1.1.54.1.43)

Number of Pause frames sent by the specified interface with the pause time greater than zero. This statistic is only available on 10G ports of the NetScaler 12000-10G platform and data ports of the NetScaler MPX 15000 and 17000 platforms. This statistic also includes the Pause frames with pause time equal to zero.

ifNicStsStalls (1.3.6.1.4.1.5951.4.1.1.54.1.44)

Number of times the status updates for a specified interface were stalled since the NetScaler appliance was started or the interface statistics were cleared. A status stall is detected when the status of the interface is not updated by the NIC hardware within 0.8 seconds of the last update.

ifNicTxStalls (1.3.6.1.4.1.5951.4.1.1.54.1.45)

Number of times the interface stalled, when transmitting packets, since the NetScaler appliance was started or the interface statistics were cleared. Transmit (Tx) stalls are detected when a packet posted for transmission is not transmitted in 4 seconds.

ifnicRxStalls (1.3.6.1.4.1.5951.4.1.1.54.1.46)

Number of times the interface stalled, when receiving packets, since the NetScaler appliance was started or the interface statistics were cleared. Receive (Rx) stalls are detected when the following conditions are met:

- (1) The link is up for more than 10 minutes.
- (2) Packets are transmitted, but no packets are received for 16 seconds.

ifnicErrDisables (1.3.6.1.4.1.5951.4.1.1.54.1.47)

Number of times the specified interface is disabled by the NetScaler, due to continuous Receive (Rx) or Transmit (Tx) stalls, since the NetScaler appliance was started or the interface statistics were cleared. The NetScaler disables an interface when one of the following conditions is met:

- (1) Three consecutive transmit stalls occur with at most a gap of 10 seconds between any two stalls.
- (2) Three consecutive receive stalls occur with at most a gap of 120 seconds between any two stalls.

ifThroughput (1.3.6.1.4.1.5951.4.1.1.54.1.48)

Interface throughput in Mbps

ifMinThroughput (1.3.6.1.4.1.5951.4.1.1.54.1.49)

Interface minimum throughput in Mbps

ifErrDroppedRxPktsRI (1.3.6.1.4.1.5951.4.1.1.54.1.50)

Number of packets dropped due to license limit on the specified interface.

ifErrRxNoNSB (1.3.6.1.4.1.5951.4.1.1.54.1.51)

Number of times the NetScaler failed to allocate buffers, for inbound packets, for the specified interface since the NetScaler appliance was started or the interface statistics were cleared.

ifInterfaceAlias (1.3.6.1.4.1.5951.4.1.1.54.1.52)

Alias Name for the Interface

nsExpressionTable (1.3.6.1.4.1.5951.4.1.1.58)

Expression related configuration

Indexed on: [expressionName](#)

expressionName (1.3.6.1.4.1.5951.4.1.1.58.1.1)

Name of the expression.

expressionTotalHits (1.3.6.1.4.1.5951.4.1.1.58.1.2)

Total number of hits to this expression.

monitorCount (1.3.6.1.4.1.5951.4.1.1.61)

Number of monitors defined on this NetScaler appliance.

monitorBindCount (1.3.6.1.4.1.5951.4.1.1.62)

Number of monitor bindings defined on this NetScaler appliance.

htmlInjectionStatsGroup (1.3.6.1.4.1.5951.4.1.1.63)

htmlInjectedBytes (1.3.6.1.4.1.5951.4.1.1.63.1)

Number of bytes injected during HTML injection.

htmlInjectMemAllocFailed (1.3.6.1.4.1.5951.4.1.1.63.4)

Number of times memory allocation failed during configuration of HTML injection.

nsSslVpnStatsGroup (1.3.6.1.4.1.5951.4.1.1.66)

indexHtmlHit (1.3.6.1.4.1.5951.4.1.1.66.1)

Number of requests for VPN login page.

indexHtmlNoServed (1.3.6.1.4.1.5951.4.1.1.66.2)

Number of failures to display VPN login page.

cfgHtmlServed (1.3.6.1.4.1.5951.4.1.1.66.3)

Number of client configuration requests received by VPN server.

dnsReqHit (1.3.6.1.4.1.5951.4.1.1.66.4)

Number of DNS queries resolved by VPN server.

winsRequestHit (1.3.6.1.4.1.5951.4.1.1.66.5)

Number of WINS queries resolved by VPN server.

csRequestHit (1.3.6.1.4.1.5951.4.1.1.66.6)

Number of SSL VPN tunnels formed between VPN server and client.

csNonHttpProbeHit (1.3.6.1.4.1.5951.4.1.1.66.7)

Number of probes from VPN to back-end non-HTTP servers that have been accessed by the VPN client.

csHttpProbeHit (1.3.6.1.4.1.5951.4.1.1.66.8)

Number of probes from VPN to back-end HTTP servers that have been accessed by the VPN client.

totalCsConnSucc (1.3.6.1.4.1.5951.4.1.1.66.9)

Number of successful probes to all back-end servers.

totalFsRequest (1.3.6.1.4.1.5951.4.1.1.66.10)

Number of file system requests received by VPN server.

iipDisabledMIPdisabled (1.3.6.1.4.1.5951.4.1.1.66.11)

Both IIP and MIP is disabled.

iipFailedMIPdisabled (1.3.6.1.4.1.5951.4.1.1.66.12)

Number of times IIP assignment failed and MIP is disabled.

iipDisabledMIPused (1.3.6.1.4.1.5951.4.1.1.66.13)

Number of times MIP is used as IIP is disabled.

iipFailedMIPused (1.3.6.1.4.1.5951.4.1.1.66.14)

Number of times MIP is used as IIP assignment failed.

socksMethReqRcvd (1.3.6.1.4.1.5951.4.1.1.66.15)

Number of received SOCKS method request.

socksMethReqSent (1.3.6.1.4.1.5951.4.1.1.66.16)

Number of sent SOCKS method request.

socksMethRespRcvd (1.3.6.1.4.1.5951.4.1.1.66.17)

Number of received SOCKS method response.

socksMethRespSent (1.3.6.1.4.1.5951.4.1.1.66.18)

Number of sent SOCKS method response.

socksConnReqRcvd (1.3.6.1.4.1.5951.4.1.1.66.19)

Number of received SOCKS connect request.

socksConnReqSent (1.3.6.1.4.1.5951.4.1.1.66.20)

Number of sent SOCKS connect request.

socksConnRespRcvd (1.3.6.1.4.1.5951.4.1.1.66.21)

Number of received SOCKS connect response.

socksConnRespSent (1.3.6.1.4.1.5951.4.1.1.66.22)

Number of sent SOCKS connect response.

socksServerError (1.3.6.1.4.1.5951.4.1.1.66.23)

Number of SOCKS server error.

socksClientError (1.3.6.1.4.1.5951.4.1.1.66.24)

Number of SOCKS client error.

staConnSuccess (1.3.6.1.4.1.5951.4.1.1.66.25)

Number of STA connection success.

staConnFailure (1.3.6.1.4.1.5951.4.1.1.66.26)

Number of STA connection failure.

cpsConnSuccess (1.3.6.1.4.1.5951.4.1.1.66.27)

Number of CPS connection success.

cpsConnFailure (1.3.6.1.4.1.5951.4.1.1.66.28)

Number of CPS connection failure.

staRequestSent (1.3.6.1.4.1.5951.4.1.1.66.29)

Number of STA request sent.

staResponseRecvd (1.3.6.1.4.1.5951.4.1.1.66.30)

Number of STA response received.

icaLicenseFailure (1.3.6.1.4.1.5951.4.1.1.66.31)

Number of ICA license failure.

staRenewSent (1.3.6.1.4.1.5951.4.1.1.66.32)

Number of STA renew requests sent.

staRenewRecvd (1.3.6.1.4.1.5951.4.1.1.66.33)

Number of STA renew response received.

staReassErr (1.3.6.1.4.1.5951.4.1.1.66.34)

Number of STA response reassembly errors.

staRnewNoCInt (1.3.6.1.4.1.5951.4.1.1.66.35)

Number of STA renew response for missing clients.

staRenewNoRfsh (1.3.6.1.4.1.5951.4.1.1.66.36)

Number of STA renew response missing refresh values.

staValidNoCInt (1.3.6.1.4.1.5951.4.1.1.66.37)

Number of STA validate response for clients that have already closed.

staValidNoEst (1.3.6.1.4.1.5951.4.1.1.66.38)

Number of STA validate responses for clients not in TCP ESTABLISHED state.

staMonSent (1.3.6.1.4.1.5951.4.1.1.66.39)

Number of STA monitor requests sent.

staMonRcvd (1.3.6.1.4.1.5951.4.1.1.66.40)

Number of STA monitor responses received.

staMonSucc (1.3.6.1.4.1.5951.4.1.1.66.41)

Number of STA monitor successful responses.

staMonFail (1.3.6.1.4.1.5951.4.1.1.66.42)

Number of STA monitor failed responses.

iipSpilloverMIPused (1.3.6.1.4.1.5951.4.1.1.66.43)

Number of times MIP is used on IIP Spillover.

IPv6toV4FindIPv6MapErr (1.3.6.1.4.1.5951.4.1.1.66.44)

Number of IPv6toIPv4 find IPv6 mapping errors.

IPv6toV4MapInsertErr (1.3.6.1.4.1.5951.4.1.1.66.45)

Number of IPv6 to IPv4 mapping Insert Errors.

parseIPv6AddressErr (1.3.6.1.4.1.5951.4.1.1.66.46)

Errors in parsing for IPv6 address from address string.

nsAaaStatsGroup (1.3.6.1.4.1.5951.4.1.1.67)**aaaAuthFail (1.3.6.1.4.1.5951.4.1.1.67.1)**

Count of authentication failures.

aaaAuthSuccess (1.3.6.1.4.1.5951.4.1.1.67.2)

Count of authentication successes.

aaaAuthNonHttpFail (1.3.6.1.4.1.5951.4.1.1.67.3)

Count of non HTTP connections that failed authorization.

aaaAuthOnlyHttpFail (1.3.6.1.4.1.5951.4.1.1.67.4)

Count of HTTP connections that failed authorization.

aaaAuthNonHttpSuccess (1.3.6.1.4.1.5951.4.1.1.67.5)

Count of non HTTP connections that succeeded authorization.

aaaAuthOnlyHttpSuccess (1.3.6.1.4.1.5951.4.1.1.67.6)

Count of HTTP connections that succeeded authorization.

aaaTotSessions (1.3.6.1.4.1.5951.4.1.1.67.7)

Count of all AAA sessions.

aaaTotSessionTimeout (1.3.6.1.4.1.5951.4.1.1.67.8)

Count of AAA sessions that have timed out.

aaaCurSessions (1.3.6.1.4.1.5951.4.1.1.67.9)

Count of current AAA sessions.

aaaCurICASessions (1.3.6.1.4.1.5951.4.1.1.67.10)

Count of current ICA only sessions.

aaaCurTMSessions (1.3.6.1.4.1.5951.4.1.1.67.11)

Count of current AAATM sessions.

aaaTotTMSessions (1.3.6.1.4.1.5951.4.1.1.67.12)

Count of all AAATM sessions.

aaaCurICAOnlyConn (1.3.6.1.4.1.5951.4.1.1.67.13)

Count of current ICA only connections.

aaaCurICAConn (1.3.6.1.4.1.5951.4.1.1.67.14)

Count of current ICA connections.

nsGlobalConfigSettings (1.3.6.1.4.1.5951.4.1.1.68)**webServerHttpPorts (1.3.6.1.4.1.5951.4.1.1.68.1)**

The HTTP ports on the Web server. System performs connection off-load for any client request that has a destination port matching one of these configured ports.

maxTcpConnections (1.3.6.1.4.1.5951.4.1.1.68.2)

The maximum number of connections that will be made from the system to the web server(s) attached to it. This value is applied globally to all attached servers.

maxReqPerConnection (1.3.6.1.4.1.5951.4.1.1.68.3)

The maximum number of requests that the system can pass on a particular connection between the system and a server attached to it. If the Value is zero, then it allows an unlimited number of requests to be passed.

ciplInsertionStatus (1.3.6.1.4.1.5951.4.1.1.68.4)

This represents the option to control (enable or disable) the insertion of the actual client IP address into the HTTP header request passed from the client to any of the servers attached to the system. The passed address can then be accessed through a minor modification to the server. If cipHeader is specified, it will be used as the client IP header. If it is not specified, then the value that has been set by the set ns config CLI command will be used as the client IP header.

ciplInsertionHeader (1.3.6.1.4.1.5951.4.1.1.68.5)

This represents the text that will be used as the client IP header.

cookieVersionInserted (1.3.6.1.4.1.5951.4.1.1.68.6)

The version of the cookie inserted by the system.

minPathMTU (1.3.6.1.4.1.5951.4.1.1.68.7)

The minimum Path MTU of the system.

mtuEntryTimeoutValue (1.3.6.1.4.1.5951.4.1.1.68.8)

The timeout value of MTU entries.

ftpPortRange (1.3.6.1.4.1.5951.4.1.1.68.9)

The Port range configured for FTP services.

nsInetAddressTable (1.3.6.1.4.1.5951.4.1.1.70)

This table contains information about the IPv6 addresses configured on the NetScaler.

Indexed on: [nsInetAddressType](#), [nsInetAddress](#)

nsInetAddressType (1.3.6.1.4.1.5951.4.1.1.70.1.1)

The address type of nsInetAddress

nsInetAddress (1.3.6.1.4.1.5951.4.1.1.70.1.2)

This represents an IPv4/v6 address configured on the NetScaler

nsInetMaskLenth (1.3.6.1.4.1.5951.4.1.1.70.1.3)

This represents netmask length.

nsInetType (1.3.6.1.4.1.5951.4.1.1.70.1.4)

This represents the IP address type

nsInetMode (1.3.6.1.4.1.5951.4.1.1.70.1.5)

This represents the IP address mode

nsInetFreePorts (1.3.6.1.4.1.5951.4.1.1.70.1.6)

This represents the number of unused ports free on this IP

nsInetVlan (1.3.6.1.4.1.5951.4.1.1.70.1.7)

The vlan to which this ip address is bound.

nsInetBridgeGroup (1.3.6.1.4.1.5951.4.1.1.70.1.8)

The bridge group to which this ip address is bound.

nsInetVxlan (1.3.6.1.4.1.5951.4.1.1.70.1.9)

The vxlan to which this ip address is bound.

nsNicStatsGroup (1.3.6.1.4.1.5951.4.1.1.71)

allNicTotRxMbits (1.3.6.1.4.1.5951.4.1.1.71.1)

Number of megabits received by the NetScaler appliance.

allNicTotTxMbits (1.3.6.1.4.1.5951.4.1.1.71.2)

Number of megabits transmitted by the NetScaler appliance.

clusterTable (1.3.6.1.4.1.5951.4.1.1.72)

The cluster table.

Indexed on: [clnodeID](#)

clnodeID (1.3.6.1.4.1.5951.4.1.1.72.1.1)

This represents the unique id of the cluster node

clPeerIP (1.3.6.1.4.1.5951.4.1.1.72.1.2)

This represents the IP address of the cluster node

clNodeIP (1.3.6.1.4.1.5951.4.1.1.72.1.3)

This represents the ip of the cluster node for which notification is being sent

clAdminState (1.3.6.1.4.1.5951.4.1.1.72.1.4)

Admin State of the node in the cluster.

clMasterState (1.3.6.1.4.1.5951.4.1.1.72.1.5)

Operational state of the cluster node.

clNodeHealth (1.3.6.1.4.1.5951.4.1.1.72.1.6)

Health of the node in the cluster.

clNodeEffectiveHealth (1.3.6.1.4.1.5951.4.1.1.72.1.7)

Health of the cluster node.

clSyncState (1.3.6.1.4.1.5951.4.1.1.72.1.8)

Sync state of the cluster node.

clNodeViewQuorum (1.3.6.1.4.1.5951.4.1.1.72.1.9)

This represents whether the node view of cluster has quorum or not

clNodeHealthReason (1.3.6.1.4.1.5951.4.1.1.72.1.10)

This represents the reason for bad health of the cluster node

nsClusterStatsGroup (1.3.6.1.4.1.5951.4.1.1.73)**clViewLeader (1.3.6.1.4.1.5951.4.1.1.73.1)**

NSIP address of the Configuration Coordinator of the cluster.

nsIcmp6StatsGroup (1.3.6.1.4.1.5951.4.1.1.76)**ipv6TotRxPkts (1.3.6.1.4.1.5951.4.1.1.76.1)**

IPv6 packets received.

ipv6TotTxPkts (1.3.6.1.4.1.5951.4.1.1.76.2)

IPv6 packets transmitted

ipv6TotRxBytes (1.3.6.1.4.1.5951.4.1.1.76.3)

Bytes of IPv6 data received.

ipv6TotTxBytes (1.3.6.1.4.1.5951.4.1.1.76.4)

Bytes of IPv6 data transmitted.

ipv6FragTotRxPkts (1.3.6.1.4.1.5951.4.1.1.76.5)

IPv6 fragments received.

ipv6FragTotPktsForward (1.3.6.1.4.1.5951.4.1.1.76.6)

IPv6 fragments forwarded to the client or server without reassembly.

ipv6FragTotPktsProcessNoReass (1.3.6.1.4.1.5951.4.1.1.76.7)

IPv6 fragments processed without reassembly.

ipv6ErrHdr (1.3.6.1.4.1.5951.4.1.1.76.8)

Packets received that contain an error in one or more components of the IPv6 header.

ipv6LandAttack (1.3.6.1.4.1.5951.4.1.1.76.9)

Land-attack packets received. The source and destination addresses are the same. If not dropped, these packets can lock up the appliance.

ipv6FragZeroLenPkt (1.3.6.1.4.1.5951.4.1.1.76.10)

Packets received with a fragment length of 0 bytes.

ipv6TotIcmpFragPkts (1.3.6.1.4.1.5951.4.1.1.76.11)

Number of ICMPV6 fragmented packets.

ipv6TotLookupDone (1.3.6.1.4.1.5951.4.1.1.76.12)

Total number of nd6 lookup done.

ipv6TotLookupFailed (1.3.6.1.4.1.5951.4.1.1.76.13)

Total number of nd6 lookup failed.

ipv6TotStaticRoutes (1.3.6.1.4.1.5951.4.1.1.76.14)

Total number of static ipv6 routes.

ipv6TotDynamicRoutes (1.3.6.1.4.1.5951.4.1.1.76.15)

Total number of dynamic ipv6 routes.

ipv6TotNeighborDiscovered (1.3.6.1.4.1.5951.4.1.1.76.16)

Total number of nd6 entries both dynamic and static.

ipv6TotIpv6To4Conversions (1.3.6.1.4.1.5951.4.1.1.76.17)

Total number of ipv6 to v4 conversion done.

ipv6TotIpv4To6Conversions (1.3.6.1.4.1.5951.4.1.1.76.18)

Total number of ipv4 to v6 conversion done.

ipv6TotTcpConnection (1.3.6.1.4.1.5951.4.1.1.76.19)

TCP connections over IPv6.

ipv6TotNonTcpConnection (1.3.6.1.4.1.5951.4.1.1.76.20)

Non TCP connections over IPv6.

nsTdlnetAddressTable (1.3.6.1.4.1.5951.4.1.1.77)

This table contains information about the non-default Td IP Addresses configured on the NetScaler.

Indexed on: [nsTdlnetId](#), [nsTdlnetAddressType](#), [nsTdlnetAddress](#)

nsTdlnetId (1.3.6.1.4.1.5951.4.1.1.77.1.1)

This represents a traffic domain ID

nsTdlnetAddressType (1.3.6.1.4.1.5951.4.1.1.77.1.2)

The address type of nsTdlnetAddress

nsTdlnetAddress (1.3.6.1.4.1.5951.4.1.1.77.1.3)

This represents an IPv4/v6 address configured on the NetScaler

nsTdlNetMaskLength (1.3.6.1.4.1.5951.4.1.1.77.1.4)

This represents netmask length.

nsTdlNetType (1.3.6.1.4.1.5951.4.1.1.77.1.5)

This represents the IP address type

nsTdlNetMode (1.3.6.1.4.1.5951.4.1.1.77.1.6)

This represents the IP address mode

nsTdlNetFreePorts (1.3.6.1.4.1.5951.4.1.1.77.1.7)

This represents the number of unused ports free on this IP

nsTdlNetVlan (1.3.6.1.4.1.5951.4.1.1.77.1.8)

The vlan to which this ip address is bound.

nsTdlNetBridgeGroup (1.3.6.1.4.1.5951.4.1.1.77.1.9)

The bridge group to which this ip address is bound.

nsTdlNetVxlan (1.3.6.1.4.1.5951.4.1.1.77.1.10)

The vxlan to which this ip address is bound.

nsCaStatsGroup (1.3.6.1.4.1.5951.4.1.1.78)**caTotlookuphit (1.3.6.1.4.1.5951.4.1.1.78.3)**

This number should be close to the number of hits being served currently.

caMsftSmthStrmVid (1.3.6.1.4.1.5951.4.1.1.78.12)

This tells the total number of MicrosoftSmoothStreaming requests served by NS

cacMsftSmthStrmVid (1.3.6.1.4.1.5951.4.1.1.78.13)

This tells the total number of MicrosoftSmoothStreaming requests served from cache

caMsftSmthStrmVidBytes (1.3.6.1.4.1.5951.4.1.1.78.14)

This tells the total number of MicrosoftSmoothStreaming bytes served by NS

caMicrosoftSmoothStreamingVidCacheBytes (1.3.6.1.4.1.5951.4.1.1.78.15)

This tells the total number of MicrosoftSmoothStreaming bytes served from cache.

caMsftSmthStrVid (1.3.6.1.4.1.5951.4.1.1.78.16)

This tells the total number of MicrosoftSmoothStreaming Playlist requests served by NS

cacMsftSmthStrmPIVid (1.3.6.1.4.1.5951.4.1.1.78.17)

This tells the total number of MicrosoftSmoothStreaming Playlist requests served from cache

caMsftSmthStrmPIVidBytes (1.3.6.1.4.1.5951.4.1.1.78.18)

This tells the total number of MicrosoftSmoothStreaming Playlist bytes served by NS

caMicrosoftSmoothStreamingPlaylistVidcacheBytes (1.3.6.1.4.1.5951.4.1.1.78.19)

This tells the total number of MicrosoftSmoothStreaming Playlist bytes served from cache

cacApleLiveStrmngVid (1.3.6.1.4.1.5951.4.1.1.78.20)

This tells the total number of AppleLive requests served by NS

cacAppleLiveStreamingVid (1.3.6.1.4.1.5951.4.1.1.78.21)

This tells the total number of AppleLive requests served from cache

caAppleLiveStreamingVidBytes (1.3.6.1.4.1.5951.4.1.1.78.22)

This tells the total number of AppleLive bytes served by NS

caAppleLiveStreamingVidcacheBytes (1.3.6.1.4.1.5951.4.1.1.78.23)

This tells the total number of AppleLive bytes served from cache

cacAppleLiveStrmngVid (1.3.6.1.4.1.5951.4.1.1.78.24)

This tells the total number of AppleLive Playlist requests served by NS

cacAppleLiveStreamingPlaylistVid (1.3.6.1.4.1.5951.4.1.1.78.25)

This tells the total number of AppleLive Playlist requests served from cache

caAppleLiveStreamingPlaylistVidBytes (1.3.6.1.4.1.5951.4.1.1.78.26)

This tells the total number of AppleLive Playlist bytes served by NS

caAppleLiveStreamingPlaylistVidcacheBytes (1.3.6.1.4.1.5951.4.1.1.78.27)

This tells the total number of AppleLive Playlist bytes served from cache

caADTSaudio (1.3.6.1.4.1.5951.4.1.1.78.28)

This tells the total number of ADTS requests served by NS

cacADTSaudio (1.3.6.1.4.1.5951.4.1.1.78.29)

This tells the total number of ADTS requests served from cache

caADTSaudioBytes (1.3.6.1.4.1.5951.4.1.1.78.30)

This tells the total number of ADTS bytes served by NS

caADTSaudiocacheBytes (1.3.6.1.4.1.5951.4.1.1.78.31)

This tells the total number of ADTS bytes served from cache

caAACaudio (1.3.6.1.4.1.5951.4.1.1.78.32)

This tells the total number of AAC requests served by NS

cacAACaudio (1.3.6.1.4.1.5951.4.1.1.78.33)

This tells the total number of AAC requests served from cache

caAACaudiobytes (1.3.6.1.4.1.5951.4.1.1.78.34)

This tells the total number of AAC bytes served by NS

caAACaudiocachebytes (1.3.6.1.4.1.5951.4.1.1.78.35)

This tells the total number of AAC bytes served from cache

caFLVid (1.3.6.1.4.1.5951.4.1.1.78.36)

This tells the total number of FLV requests served by NS

cacFLVid (1.3.6.1.4.1.5951.4.1.1.78.37)

This tells the total number of FLV requests served from cache

caFLVidBytes (1.3.6.1.4.1.5951.4.1.1.78.38)

This tells the total number of FLV bytes served by NS

caFLVVidcacheBytes (1.3.6.1.4.1.5951.4.1.1.78.39)

This tells the total number of FLV bytes served from cache

caMP4Vid (1.3.6.1.4.1.5951.4.1.1.78.40)

This tells the total number of MP4 requests served by NS

cacMP4Vid (1.3.6.1.4.1.5951.4.1.1.78.41)

This tells the total number of MP4 requests served from cache

caMP4VidBytes (1.3.6.1.4.1.5951.4.1.1.78.42)

This tells the total number of MP4 bytes served by NS

caMP4VidcacheBytes (1.3.6.1.4.1.5951.4.1.1.78.43)

This tells the total number of MP4 bytes served from cache

ca3PVid (1.3.6.1.4.1.5951.4.1.1.78.44)

This tells the total number of 3GP requests served by NS

ca3GPVid (1.3.6.1.4.1.5951.4.1.1.78.45)

This tells the hit ratio of 3GP requests served from cache

ca3GPVidBytes (1.3.6.1.4.1.5951.4.1.1.78.46)

This tells the total number of 3GP bytes served by NS

ca3GPVidcacheBytes (1.3.6.1.4.1.5951.4.1.1.78.47)

This tells the total number of 3GP bytes served from cache

caMsftSmthStrmVidHR (1.3.6.1.4.1.5951.4.1.1.78.48)

This tells the hit ratio of MicrosoftSmoothStreaming requests

cacMsftSmthStrmPIVidHR (1.3.6.1.4.1.5951.4.1.1.78.49)

This tells the hit ratio of MicrosoftSmoothStreaming Playlist requests

cacAppleLiveStreamingVidHR (1.3.6.1.4.1.5951.4.1.1.78.50)

This tells the hit ratio of AppleLive requests

caAppleLiveStreamingPlaylistVidHR (1.3.6.1.4.1.5951.4.1.1.78.51)

This tells the hit ratio of AppleLive Playlist requests

cacADTSaudioHR (1.3.6.1.4.1.5951.4.1.1.78.52)

This tells the Hit Ratio of ADTS requests

caAACaudioHR (1.3.6.1.4.1.5951.4.1.1.78.53)

This tells the hit ratio of AAC requests

caFLVVidHR (1.3.6.1.4.1.5951.4.1.1.78.54)

This tells the hit ratio of FLV requests

cacMP4VidHR (1.3.6.1.4.1.5951.4.1.1.78.55)

This tells the hit ratio of MP4 requests

ca3GPVidHR (1.3.6.1.4.1.5951.4.1.1.78.56)

This tells the total number of 3GP requests

caMsftSmthStrmngVidCaBytesHR (1.3.6.1.4.1.5951.4.1.1.78.57)

This tells the Bytes hit ratio of MicrosoftSmoothStreaming bytes .

caMsftSmthStrmngPIVidcaBytesHR (1.3.6.1.4.1.5951.4.1.1.78.58)

This tells the byte hit ratio of MicrosoftSmoothStreaming Playlist bytes

caAppleLiveStrmngVidcacheBytesHR (1.3.6.1.4.1.5951.4.1.1.78.59)

This tells the total number of AppleLive bytes

caAppleLiveStrmngPIVidcacheBytesHR (1.3.6.1.4.1.5951.4.1.1.78.60)

This tells the byte hit ratio of AppleLive Playlist bytes

caADTSaudiocacheBytesHR (1.3.6.1.4.1.5951.4.1.1.78.61)

This tells the byte hit ratio of ADTS bytes

caAACaudiocachebytesHR (1.3.6.1.4.1.5951.4.1.1.78.62)

This tells the hit ratio AAC bytes served

caFLVvidcacheBytesHR (1.3.6.1.4.1.5951.4.1.1.78.63)

This tells the hit ratio of FLV bytes

caMP4VidcacheBytesHR (1.3.6.1.4.1.5951.4.1.1.78.64)

This tells the hit ratio of MP4 bytes

ca3GPVidcacheBytesHR (1.3.6.1.4.1.5951.4.1.1.78.65)

This tells the hit ratio of 3GP bytes

caAndroid (1.3.6.1.4.1.5951.4.1.1.78.66)

Total number of android requests to netscaler

caLaptopDesktop (1.3.6.1.4.1.5951.4.1.1.78.67)

Total number of laptop/desktop requests to netscaler

calos (1.3.6.1.4.1.5951.4.1.1.78.68)

Total number of iOS requests to netscaler

caOther (1.3.6.1.4.1.5951.4.1.1.78.69)

Total number of other mobile device requests to netscaler

caUnidentified (1.3.6.1.4.1.5951.4.1.1.78.70)

Total number of unidentified requests to netscaler

caAndroidcache (1.3.6.1.4.1.5951.4.1.1.78.71)

This tells android requests served from cache

caloscache (1.3.6.1.4.1.5951.4.1.1.78.72)

This tells iOS requests served from cache

caOthercache (1.3.6.1.4.1.5951.4.1.1.78.73)

This tells Other device requests served from cache

calaptopDesktopcache (1.3.6.1.4.1.5951.4.1.1.78.74)

This tells laptop/desktop requests served from cache

caUnidentifiedcache (1.3.6.1.4.1.5951.4.1.1.78.75)

This tells unidentified device requests served from cache

caAndroidBytes (1.3.6.1.4.1.5951.4.1.1.78.76)

This tells the total number of Android bytes served by NS

calosBytes (1.3.6.1.4.1.5951.4.1.1.78.77)

This tells the total number of IOS bytes served by NS

caOtherBytes (1.3.6.1.4.1.5951.4.1.1.78.78)

This tells the total number of Other mobile device bytes served by NS

caAlptopDsktpBytes (1.3.6.1.4.1.5951.4.1.1.78.79)

This tells the total number of Laptop/desktop bytes served by NS

caUnidentifiedBytes (1.3.6.1.4.1.5951.4.1.1.78.80)

This tells the total number of unidentified device bytes served by NS

caAndroididcacheBytes (1.3.6.1.4.1.5951.4.1.1.78.81)

This tells the total number of Android bytes served from cache

calosidcacheBytes (1.3.6.1.4.1.5951.4.1.1.78.82)

This tells the total number of IOS bytes served from cache

caOtherididcacheBytes (1.3.6.1.4.1.5951.4.1.1.78.83)

This tells the total number of other device bytes served from cache

caLaptpdsktpBytes (1.3.6.1.4.1.5951.4.1.1.78.84)

This tells the total number of Laptop/desktop bytes served from cache

caAunidentifiedBytes (1.3.6.1.4.1.5951.4.1.1.78.85)

This tells the total number of unidentified device bytes served from cache

caAndroidHR (1.3.6.1.4.1.5951.4.1.1.78.86)

This tells the hit ratio of android requests

calaptopDesktphr (1.3.6.1.4.1.5951.4.1.1.78.87)

This tells the hit ratio of laptop/desktop requests

caotherhr (1.3.6.1.4.1.5951.4.1.1.78.88)

This tells the hit ratio of other mobile device requests

caloshr (1.3.6.1.4.1.5951.4.1.1.78.89)

This tells the hit ratio of ios requests

caUnidentifiedhr (1.3.6.1.4.1.5951.4.1.1.78.90)

This tells the hit ratio of android requests

caAndroidByteshr (1.3.6.1.4.1.5951.4.1.1.78.91)

This tells the hit ratio of 3GP bytes

caLaptpdsktpByteshr (1.3.6.1.4.1.5951.4.1.1.78.92)

This tells the hit ratio of laptop_desktop bytes

caotherBytesHR (1.3.6.1.4.1.5951.4.1.1.78.93)

This tells the hit ratio of Other device

caiosBytesHR (1.3.6.1.4.1.5951.4.1.1.78.94)

This tells the hit ratio of IOS bytes

caAunidentifiedBytesHR (1.3.6.1.4.1.5951.4.1.1.78.95)

This tells the hit ratio of unidentified bytes

nsvPathStatsGroup (1.3.6.1.4.1.5951.4.1.1.79)

vPathTotL2DataRx (1.3.6.1.4.1.5951.4.1.1.79.1)

Total number of non-fragmented vPath data packets decapsulated in L2 adjacency

vPathTotL3DataRx (1.3.6.1.4.1.5951.4.1.1.79.2)

Total number of non-fragmented vPath data packets decapsulated in L3 adjacency

vPathTotL2CntrlPkts (1.3.6.1.4.1.5951.4.1.1.79.3)

Total number of vPath control packets received in L2 adjacency

vPathTotL3CntrlPkts (1.3.6.1.4.1.5951.4.1.1.79.4)

Total number of vPath control packets received in L3 adjacency

vPathTotFragPkts (1.3.6.1.4.1.5951.4.1.1.79.5)

Total number of vPath fragments received

vPathTotL2EncapPkts (1.3.6.1.4.1.5951.4.1.1.79.6)

Total number of L2 vPath encapsulated packets injected to VEM

vPathTotL3EncapPkts (1.3.6.1.4.1.5951.4.1.1.79.7)

Total number of L3 vPath encapsulated packets injected to VEM

vPathTotFragEncapPkts (1.3.6.1.4.1.5951.4.1.1.79.8)

Number of fragmented vPath packets transmitted

vPathTotOffload (1.3.6.1.4.1.5951.4.1.1.79.9)

Number of offloaded vPath packets transmitted

vxlanTable (1.3.6.1.4.1.5951.4.1.1.81)

The vxlan related statistics table.

Indexed on: [vxlanVNid](#)

vxlanVNid (1.3.6.1.4.1.5951.4.1.1.81.1.1)

This represents the VNID of the vxlan

vxlanTotRxPkts (1.3.6.1.4.1.5951.4.1.1.81.1.2)

Packets received on the VXLAN.

vxlanTotRxBytes (1.3.6.1.4.1.5951.4.1.1.81.1.3)

Bytes of data received on the VXLAN.

vxlanTotTxPkts (1.3.6.1.4.1.5951.4.1.1.81.1.4)

Packets transmitted on the VXLAN.

vxlanTotTxBytes (1.3.6.1.4.1.5951.4.1.1.81.1.5)

Bytes of data transmitted on the VXLAN.

cacheGroupTable (1.3.6.1.4.1.5951.4.1.1.82)

Content Group Table

Indexed on: **cachegroupName**

cachegroupName (1.3.6.1.4.1.5951.4.1.1.82.1.1)

Encoded name of the cache group

groupnon304hit (1.3.6.1.4.1.5951.4.1.1.82.1.2)

Non304 hits for ContentGroup

group304hit (1.3.6.1.4.1.5951.4.1.1.82.1.3)

304 hits for ContentGroup

totcell (1.3.6.1.4.1.5951.4.1.1.82.1.4)

Number of objects in contentgroup

totmarkercell (1.3.6.1.4.1.5951.4.1.1.82.1.5)

Number of marker objects in contentgroup

timesflushed (1.3.6.1.4.1.5951.4.1.1.82.1.6)

Number of times contentgroup is flushed

totmemory (1.3.6.1.4.1.5951.4.1.1.82.1.7)

current memory usage

maxmemory (1.3.6.1.4.1.5951.4.1.1.82.1.8)

maximum memory usage limit

aclStatsGroup (1.3.6.1.4.1.5951.4.1.1.22.1)

aclTotPktsBridged (1.3.6.1.4.1.5951.4.1.1.22.1.9)

Packets matching a bridge ACL, which is in transparent mode and bypasses service processing.

aclTotPktsDenied (1.3.6.1.4.1.5951.4.1.1.22.1.10)

Packets dropped because they match ACLs with processing mode set to DENY.

aclTotPktsAllowed (1.3.6.1.4.1.5951.4.1.1.22.1.11)

Packets matching ACLs with processing mode set to ALLOW. NetScaler processes these packets.

aclTotHits (1.3.6.1.4.1.5951.4.1.1.22.1.12)

Packets matching an ACL.

aclTotMisses (1.3.6.1.4.1.5951.4.1.1.22.1.13)

Packets not matching any ACL.

aclTotPktsNAT (1.3.6.1.4.1.5951.4.1.1.22.1.14)

Packets matching a NAT ACL, resulting in a NAT session.

nsAcITable (1.3.6.1.4.1.5951.4.1.1.22.1.20)

This table contains all the ACLs configured

Indexed on: [aclName](#)

aclName (1.3.6.1.4.1.5951.4.1.1.22.1.20.1.1)

The name of the ACL

aclPriority (1.3.6.1.4.1.5951.4.1.1.22.1.20.1.2)

The priority of the ACL

aclperHits (1.3.6.1.4.1.5951.4.1.1.22.1.20.1.4)

Number of times the acl was hit

aclFullName (1.3.6.1.4.1.5951.4.1.1.22.1.20.1.5)

The full name of the ACL

aclTotCount (1.3.6.1.4.1.5951.4.1.1.22.1.21)

Total number of ACL rules configured.

sacStatsGroup (1.3.6.1.4.1.5951.4.1.1.22.3)

sacTotPktsBridged (1.3.6.1.4.1.5951.4.1.1.22.3.1)

Total packets that matched a SimpleACL with action BRIDGE and got bridged by NetScaler.

sacTotPktsDenied (1.3.6.1.4.1.5951.4.1.1.22.3.2)

Packets dropped because they match SimpleACL (Access Control List) with processing mode set to DENY.

sacTotPktsAllowed (1.3.6.1.4.1.5951.4.1.1.22.3.3)

Total packets that matched a SimpleACL with action ALLOW and got consumed by NetScaler.

sacTotHits (1.3.6.1.4.1.5951.4.1.1.22.3.4)

Packets matching a SimpleACL.

sacTotMisses (1.3.6.1.4.1.5951.4.1.1.22.3.5)

Packets not matching any SimpleACL.

sacIsCount (1.3.6.1.4.1.5951.4.1.1.22.3.6)

Number of SimpleACLs configured.

acl6StatsGroup (1.3.6.1.4.1.5951.4.1.1.22.4)

nsAcl6Table (1.3.6.1.4.1.5951.4.1.1.22.4.20)

This table contains all the ACLs6 configured

Indexed on: [acAclName](#)

acAclName (1.3.6.1.4.1.5951.4.1.1.22.4.20.1.1)

The name of the ACL6

acl6Priority (1.3.6.1.4.1.5951.4.1.1.22.4.20.1.2)

The priority of the ACL6

acl6perHits (1.3.6.1.4.1.5951.4.1.1.22.4.20.1.3)

Number of times the acl6 was hit

acl6FullName (1.3.6.1.4.1.5951.4.1.1.22.4.20.1.4)

The full name of the ACL6

acl6TotPktsBridged (1.3.6.1.4.1.5951.4.1.1.22.4.21)

Packets matching a bridge IPv6 ACL, which is in transparent mode and bypasses service processing.

acl6TotPktsDenied (1.3.6.1.4.1.5951.4.1.1.22.4.22)

Packets dropped because they match IPv6 ACLs with processing mode set to DENY.

acl6TotPktsAllowed (1.3.6.1.4.1.5951.4.1.1.22.4.23)

Packets matching IPv6 ACLs with processing mode set to ALLOW. NetScaler processes these packets.

acl6TotPktsNAT (1.3.6.1.4.1.5951.4.1.1.22.4.24)

Packets matching a NAT ACL6, resulting in a NAT session.

acl6TotHits (1.3.6.1.4.1.5951.4.1.1.22.4.25)

Packets matching an IPv6 ACL.

acl6TotMisses (1.3.6.1.4.1.5951.4.1.1.22.4.26)

Packets not matching any IPv6 ACL.

acl6TotPktsNAT64 (1.3.6.1.4.1.5951.4.1.1.22.4.27)

Packets matching a NAT64 ACL6, resulting in a NAT64 translation.

acl6TotCount (1.3.6.1.4.1.5951.4.1.1.22.4.28)

Total number of ACL6 rules configured.

pbrStatsGroup (1.3.6.1.4.1.5951.4.1.1.22.5)**nsPbrTable (1.3.6.1.4.1.5951.4.1.1.22.5.20)**

This table contains all the PBRs configured

Indexed on: [pbrName](#)

pbrName (1.3.6.1.4.1.5951.4.1.1.22.5.20.1.1)

The name of the PBR

pbrFullName (1.3.6.1.4.1.5951.4.1.1.22.5.20.1.2)

The Full name of the PBR

pbrPriority (1.3.6.1.4.1.5951.4.1.1.22.5.20.1.3)

The priority of the PBR

pbrperHits (1.3.6.1.4.1.5951.4.1.1.22.5.20.1.4)

Number of times the pbr was hit

pbrTotPktsAllowed (1.3.6.1.4.1.5951.4.1.1.22.5.21)

Total packets that matched the PBR (Policy-Based Routes) with action ALLOW

pbrTotPktsDenied (1.3.6.1.4.1.5951.4.1.1.22.5.22)

Total packets that matched the PBR with action DENY

pbrTotHits (1.3.6.1.4.1.5951.4.1.1.22.5.23)

Total packets that matched one of the configured PBR

pbrTotMisses (1.3.6.1.4.1.5951.4.1.1.22.5.24)

Total packets that did not match any PBR

sac16StatsGroup (1.3.6.1.4.1.5951.4.1.1.22.6)**sac16TotPktsBridged (1.3.6.1.4.1.5951.4.1.1.22.6.1)**

Total packets that matched a SimpleACL6 with action BRIDGE and got bridged by NetScaler.

sac16TotPktsDenied (1.3.6.1.4.1.5951.4.1.1.22.6.2)

Packets dropped because they match SimpleACL6 with processing mode set to DENY.

sac16TotPktsAllowed (1.3.6.1.4.1.5951.4.1.1.22.6.3)

Total packets that matched a SimpleACL6 with action ALLOW and got consumed by NetScaler.

sac16TotHits (1.3.6.1.4.1.5951.4.1.1.22.6.4)

Packets matching a SimpleACL6.

sac16TotMisses (1.3.6.1.4.1.5951.4.1.1.22.6.5)

Packets not matching any SimpleACL6.

sac16sCount (1.3.6.1.4.1.5951.4.1.1.22.6.6)

Number of SimpleACL6s configured.

pbr6StatsGroup (1.3.6.1.4.1.5951.4.1.1.22.7)**nsPbr6Table (1.3.6.1.4.1.5951.4.1.1.22.7.20)**

This table contains all the PBRs configured

Indexed on: [acPbrName](#)

acPbrName (1.3.6.1.4.1.5951.4.1.1.22.7.20.1.1)

The name of the PBR6

pbr6FullName (1.3.6.1.4.1.5951.4.1.1.22.7.20.1.2)

The full name of the PBR6

pbr6Priority (1.3.6.1.4.1.5951.4.1.1.22.7.20.1.3)

The priority of the PBR6

pbr6perHits (1.3.6.1.4.1.5951.4.1.1.22.7.20.1.4)

Number of times the pbr6 was hit

pbr6TotPktsAllowed (1.3.6.1.4.1.5951.4.1.1.22.7.21)

Total packets that matched the PBR6 with action ALLOW

pbr6TotPktsDenied (1.3.6.1.4.1.5951.4.1.1.22.7.22)

Total packets that matched PBR6 with action DENY

pbr6TotHits (1.3.6.1.4.1.5951.4.1.1.22.7.23)

Total packets that matched one of the configured PBR6

pbr6TotMisses (1.3.6.1.4.1.5951.4.1.1.22.7.24)

Total packets that did not match any PBR6

gslbGlobalStats (1.3.6.1.4.1.5951.4.1.1.51.1)

customEntries (1.3.6.1.4.1.5951.4.1.1.51.1.1)

This is the number of custom locations

staticEntries (1.3.6.1.4.1.5951.4.1.1.51.1.2)

This is the number of static locations

nsPolicyStatsTable (1.3.6.1.4.1.5951.4.1.1.52.1)

This table contains the statistics for all policies

Indexed on: [pengPolicyName](#)

pengPolicyName (1.3.6.1.4.1.5951.4.1.1.52.1.1.1)

Encoded name of the policy

pengPolicyHits (1.3.6.1.4.1.5951.4.1.1.52.1.1.2)

Total policy hits count

pengBytesIn (1.3.6.1.4.1.5951.4.1.1.52.1.1.3)

Input traffic of a compression policy

pengBytesOut (1.3.6.1.4.1.5951.4.1.1.52.1.1.4)

Output traffic of a compression policy

pengPolicyFullName (1.3.6.1.4.1.5951.4.1.1.52.1.1.5)

Full name of the policy

nsDnsServerStatsGroup (1.3.6.1.4.1.5951.4.1.1.53.1)

dnsTotQueries (1.3.6.1.4.1.5951.4.1.1.53.1.1)

Total number of DNS queries received.

dnsTotAnswers (1.3.6.1.4.1.5951.4.1.1.53.1.2)

Total number of DNS responses received.

dnsTotUnsupportedResponseClass (1.3.6.1.4.1.5951.4.1.1.53.1.13)

Total number of responses for which response types were unsupported.

dnsTotUnsupportedResponseType (1.3.6.1.4.1.5951.4.1.1.53.1.14)

Total number of responses for which response type requested was unsupported.

dnsTotUnsupportedQueries (1.3.6.1.4.1.5951.4.1.1.53.1.15)

Total number of requests for which query type requested was unsupported.

dnsTotUnsupportedQueryClass (1.3.6.1.4.1.5951.4.1.1.53.1.16)

Total number of queries for which query class was unsupported.

dnsTotInvalidQueryFormat (1.3.6.1.4.1.5951.4.1.1.53.1.17)

Total number of queries whose format was invalid.

dnsTotNonAuthNoDdatas (1.3.6.1.4.1.5951.4.1.1.53.1.18)

Total number of responses for which there was a format error.

dnsTotMultiQuery (1.3.6.1.4.1.5951.4.1.1.53.1.19)

Total number of Multi Query request received.

dnsTotStrayAnswer (1.3.6.1.4.1.5951.4.1.1.53.1.20)

Total number of stray answers.

dnsTotCacheFlush (1.3.6.1.4.1.5951.4.1.1.53.1.21)

Total number of times cache was flushed.

dnsTotCacheEntriesFlush (1.3.6.1.4.1.5951.4.1.1.53.1.22)

Total number of cache entries flushed.

dnsTotServerQuery (1.3.6.1.4.1.5951.4.1.1.53.1.23)

Total number of Server queries sent.

dnsTotServerResponse (1.3.6.1.4.1.5951.4.1.1.53.1.24)

Total number of Server responses received.

dnsTotRecUpdate (1.3.6.1.4.1.5951.4.1.1.53.1.34)

Total number of record updates.

dnsTotMultiQueryDisableError (1.3.6.1.4.1.5951.4.1.1.53.1.35)

Total number of times a multi query was disabled and received a multi query.

dnsCurAuthEntries (1.3.6.1.4.1.5951.4.1.1.53.1.41)

Total number of authoritative entries.

dnsCurNoAuthEntries (1.3.6.1.4.1.5951.4.1.1.53.1.42)

Total number of non-authoritative entries.

dnsTotAuthAns (1.3.6.1.4.1.5951.4.1.1.53.1.43)

Number of queries which were authoritatively answered.

dnsTotAuthNoNames (1.3.6.1.4.1.5951.4.1.1.53.1.44)

Number of queries for which no record was found.

dnsTotNoDataResps (1.3.6.1.4.1.5951.4.1.1.53.1.45)

Number of DNS responses received without answer.

dnsTotResponseBadLen (1.3.6.1.4.1.5951.4.1.1.53.1.46)

Number of DNS responses received with invalid resource data length.

dnsTotReqRefusals (1.3.6.1.4.1.5951.4.1.1.53.1.47)

Number of DNS requests refused.

dnsTotOtherErrors (1.3.6.1.4.1.5951.4.1.1.53.1.48)

Total number of other errors.

dns64TotQueries (1.3.6.1.4.1.5951.4.1.1.53.1.68)

Total number of DNS64 queries received.

dns64TotAnswers (1.3.6.1.4.1.5951.4.1.1.53.1.69)

Total number of DNS64 answers served.

dns64TotSvrAQueries (1.3.6.1.4.1.5951.4.1.1.53.1.70)

Total number of Queries sent by DNS64 module to backend.

dnsErrNullAttack (1.3.6.1.4.1.5951.4.1.1.53.1.71)

Total number of queries received where all the counts are 0.

nsdnsRegisterTable (1.3.6.1.4.1.5951.4.1.1.53.2)

This table contains statistics about each DNS record type

Indexed on: [dnsRecordType](#)

dnsRecordType (1.3.6.1.4.1.5951.4.1.1.53.2.1.1)

DNS record type

dnsTotEntries (1.3.6.1.4.1.5951.4.1.1.53.2.1.2)

Total number of DNS record entries

dnsTotUpdates (1.3.6.1.4.1.5951.4.1.1.53.2.1.3)

Total number of DNS proactive updates

dnsTotResponses (1.3.6.1.4.1.5951.4.1.1.53.2.1.4)

Total number of DNS server responses

dnsTotRequests (1.3.6.1.4.1.5951.4.1.1.53.2.1.5)

Total number of DNS queries recieved

dnsTotErrLimits (1.3.6.1.4.1.5951.4.1.1.53.2.1.6)

Total number of times we have recieved dns record with more entries than we support

dnsTotErrRespForm (1.3.6.1.4.1.5951.4.1.1.53.2.1.7)

Total number of times we have recieved malformed responses from the backend

dnsTotErrAliasEx (1.3.6.1.4.1.5951.4.1.1.53.2.1.8)

Total number of times we have recieved non-cname record for a domain for which an alias exists

dnsTotErrNoDomains (1.3.6.1.4.1.5951.4.1.1.53.2.1.9)

Total number of cache misses

dnsCurEntries (1.3.6.1.4.1.5951.4.1.1.53.2.1.10)

Current number of DNS entries

dnsCurRecords (1.3.6.1.4.1.5951.4.1.1.53.2.1.11)

Current number of DNS Records

scPolicyStatistics (1.3.6.1.4.1.5951.4.1.1.55.1)

scPolicyUrlHits (1.3.6.1.4.1.5951.4.1.1.55.1.1)

Total number of incoming requests that matched configured sureconnect policies.

scPopUps (1.3.6.1.4.1.5951.4.1.1.55.1.2)

Total number of in-memory java script served which throws the pop-up window.

scAltContUrls (1.3.6.1.4.1.5951.4.1.1.55.1.3)

Total number of alternate content served which throws the pop-up window.

scSessionReqs (1.3.6.1.4.1.5951.4.1.1.55.1.4)

Total number of requests that were handled in a single SureConnect session.

scPostReqs (1.3.6.1.4.1.5951.4.1.1.55.1.5)

Total number of HTTP POST requests that triggered SureConnect feature.

scThresholdFail (1.3.6.1.4.1.5951.4.1.1.55.1.6)

Total number of times SureConnect was not triggered because the thresholds conditions failed.

scFaultyCookies (1.3.6.1.4.1.5951.4.1.1.55.1.7)

Total number of corrupted SureConnect cookies.

scUnSupBrow (1.3.6.1.4.1.5951.4.1.1.55.1.8)

Total number of requests that came from all unsupported browsers.

scResetStats (1.3.6.1.4.1.5951.4.1.1.55.1.9)

Total number of times that SureConnect statistics were reset.

scTotCondTriggered (1.3.6.1.4.1.5951.4.1.1.55.1.10)

Number of times that SureConnect conditions were triggered.

scTotReissuedRequests (1.3.6.1.4.1.5951.4.1.1.55.1.11)

Total number of reissued SureConnect requests.

sslCertKeyTable (1.3.6.1.4.1.5951.4.1.1.56.1)

The ssl certificate key pair configuration information

Indexed on: [sslCertKeyName](#)

sslCertKeyName (1.3.6.1.4.1.5951.4.1.1.56.1.1.1)

The certificate key pair Name.

sslCertPath (1.3.6.1.4.1.5951.4.1.1.56.1.1.2)

The certificate path.

sslKeyPath (1.3.6.1.4.1.5951.4.1.1.56.1.1.3)

The private key path.

sslInputFormat (1.3.6.1.4.1.5951.4.1.1.56.1.1.4)

The input format of the certificate key pair.

sslDaysToExpire (1.3.6.1.4.1.5951.4.1.1.56.1.1.5)

Number of days remaining for the certificate to expire.

sslCrITable (1.3.6.1.4.1.5951.4.1.1.56.2)

The ssl CRL configuration information

Indexed on: [sslCrIName](#)

sslCrIName (1.3.6.1.4.1.5951.4.1.1.56.2.1.1)

The name of CRL.

sslCrIPath (1.3.6.1.4.1.5951.4.1.1.56.2.1.2)

The CRL path.

sslCrlInputFormat (1.3.6.1.4.1.5951.4.1.1.56.2.1.3)

The input format of CRL.

sslCipherGroupTable (1.3.6.1.4.1.5951.4.1.1.56.3)

The Cipher group configuration information

Indexed on: [sslCipherGroupName](#), [sslCipherName](#)

sslCipherGroupName (1.3.6.1.4.1.5951.4.1.1.56.3.1.1)

The Cipher group name.

sslCipherName (1.3.6.1.4.1.5951.4.1.1.56.3.1.2)

The Cipher name.

sslCipherDesc (1.3.6.1.4.1.5951.4.1.1.56.3.1.3)

The Cipher description.

dosPolicyTable (1.3.6.1.4.1.5951.4.1.1.57.1)

The dos policy configuration table

Indexed on: [dosPolicyName](#)

dosPolicyName (1.3.6.1.4.1.5951.4.1.1.57.1.1.1)

Name of the policy

thresholdValue (1.3.6.1.4.1.5951.4.1.1.57.1.1.2)

Threshold surge count to detect an attack, for this DosPolicy

dosPolicyStatistics (1.3.6.1.4.1.5951.4.1.1.57.2)

dosTotConditionTriggered (1.3.6.1.4.1.5951.4.1.1.57.2.1)

Number of times the NetScaler appliance triggered the DOS JavaScript due to a condition match.

dosTotValidCookies (1.3.6.1.4.1.5951.4.1.1.57.2.2)

Number of clients from whom the NetScaler appliance received a valid DOS cookie.

dosTotDosPriorityClients (1.3.6.1.4.1.5951.4.1.1.57.2.3)

Number of valid clients that were given DOS priority.

dosAvgValidClients (1.3.6.1.4.1.5951.4.1.1.57.2.4)

Average number of DOS clients that returned a valid DOS cookie.

dosAvgDospriClients (1.3.6.1.4.1.5951.4.1.1.57.2.5)

Average number of clients that were given DOS priority.

pqPolicyConfigTable (1.3.6.1.4.1.5951.4.1.1.59.1)

The priority queuing policy configuration table

Indexed on: [pqName](#)

pqName (1.3.6.1.4.1.5951.4.1.1.59.1.1.1)

The name of the PQ policy bound to Load Balancing vserver.

pqRuleName (1.3.6.1.4.1.5951.4.1.1.59.1.1.2)

The Rule Name configuration for PQ policy.

pqQdepthThreshold (1.3.6.1.4.1.5951.4.1.1.59.1.1.3)

The vip threshold value for qdepth in PQ policy.

pqPolQdepthThreshold (1.3.6.1.4.1.5951.4.1.1.59.1.1.4)

The policy threshold value for qdepth in PQ policy.

pqPriority (1.3.6.1.4.1.5951.4.1.1.59.1.1.5)

The priority of this PQ policy.

pqPolicyWeight (1.3.6.1.4.1.5951.4.1.1.59.1.1.6)

The weight of this PQ policy.

pqPolicyStatistics (1.3.6.1.4.1.5951.4.1.1.59.2)**pqTotalPolicyMatches (1.3.6.1.4.1.5951.4.1.1.59.2.1)**

Number of times the Netscaler appliance matched an incoming request using any priority queuing policy.

pqTotalThresholdFailed (1.3.6.1.4.1.5951.4.1.1.59.2.2)

Number of times the Netscaler appliance failed to match an incoming request to any of priority queuing policy.

pqPriority1Requests (1.3.6.1.4.1.5951.4.1.1.59.2.3)

Number of priority 1 requests that the Netscaler appliance received.

pqPriority2Requests (1.3.6.1.4.1.5951.4.1.1.59.2.4)

Number of priority 2 requests that the Netscaler appliance received.

pqPriority3Requests (1.3.6.1.4.1.5951.4.1.1.59.2.5)

Number of priority 3 requests that the Netscaler appliance received.

crPolicyMapConfigTable (1.3.6.1.4.1.5951.4.1.1.60.1)

The CR map configuration information

Indexed on: [crMapName](#)

crMapName (1.3.6.1.4.1.5951.4.1.1.60.1.1.1)

The name of the map policy.

crMapSrcName (1.3.6.1.4.1.5951.4.1.1.60.1.1.2)

The name of the source domain hosted by the map.

crMapDstName (1.3.6.1.4.1.5951.4.1.1.60.1.1.3)

The name of the destination domain hosted by the map.

crMapSrcUrl (1.3.6.1.4.1.5951.4.1.1.60.1.1.4)

The Url to be modified under the given source domain.

crMapDstUrl (1.3.6.1.4.1.5951.4.1.1.60.1.1.5)

The Url after mapping.

appFirewallStatistics (1.3.6.1.4.1.5951.4.1.1.64.1)**appFirewallRequests (1.3.6.1.4.1.5951.4.1.1.64.1.1)**

HTTP/HTTPS requests sent to your protected web servers via the Application Firewall.

appFirewallResponses (1.3.6.1.4.1.5951.4.1.1.64.1.2)

HTTP/HTTPS responses sent by your protected web servers via the Application Firewall.

appFirewallAborts (1.3.6.1.4.1.5951.4.1.1.64.1.3)

Incomplete HTTP/HTTPS requests aborted by the client before the Application Firewall could finish processing them.

appFirewallRedirects (1.3.6.1.4.1.5951.4.1.1.64.1.4)

HTTP/HTTPS requests redirected by the Application Firewall to a different Web page or web server. (HTTP 302)

appFirewallViolStartURL (1.3.6.1.4.1.5951.4.1.1.64.1.5)

Number of Start URL security check violations seen by the Application Firewall.

appFirewallViolDenyURL (1.3.6.1.4.1.5951.4.1.1.64.1.6)

Number of Deny URL security check violations seen by the Application Firewall.

appFirewallViolBufferOverflow (1.3.6.1.4.1.5951.4.1.1.64.1.7)

Number of Buffer Overflow security check violations seen by the Application Firewall.

appFirewallViolCookie (1.3.6.1.4.1.5951.4.1.1.64.1.8)

Number of Cookie Consistency security check violations seen by the Application Firewall.

appFirewallViolXSS (1.3.6.1.4.1.5951.4.1.1.64.1.9)

Number of HTML Cross-Site Scripting security check violations seen by the Application Firewall.

appFirewallViolSQL (1.3.6.1.4.1.5951.4.1.1.64.1.10)

Number of HTML SQL Injection security check violations seen by the Application Firewall.

appFirewallViolFieldformat (1.3.6.1.4.1.5951.4.1.1.64.1.11)

Number of Field Format security check violations seen by the Application Firewall.

appFirewallViolFieldConsistency (1.3.6.1.4.1.5951.4.1.1.64.1.12)

Number of Field Consistency security check violations seen by the Application Firewall.

appFirewallViolCreditCard (1.3.6.1.4.1.5951.4.1.1.64.1.13)

Number of Credit Card security check violations seen by the Application Firewall.

appFirewallViolSafeObject (1.3.6.1.4.1.5951.4.1.1.64.1.14)

Number of Safe Object security check violations seen by the Application Firewall.

appFirewallTotalViol (1.3.6.1.4.1.5951.4.1.1.64.1.15)

Total number of security check violations seen by the Application Firewall.

appFirewallViolWellformednessViolations (1.3.6.1.4.1.5951.4.1.1.64.1.16)

Number of XML Format security check violations seen by the Application Firewall.

appFirewallViolXdosViolations (1.3.6.1.4.1.5951.4.1.1.64.1.17)

Number of XML Denial-of-Service security check violations seen by the Application Firewall.

appFirewallViolMsgValViolations (1.3.6.1.4.1.5951.4.1.1.64.1.18)

Number of XML Message Validation security check violations seen by the Application Firewall.

appFirewallViolWSIViolations (1.3.6.1.4.1.5951.4.1.1.64.1.19)

Number of Web Services Interoperability (WS-I) security check violations seen by the Application Firewall.

appFirewallViolXmlSqlViolations (1.3.6.1.4.1.5951.4.1.1.64.1.20)

Number of XML SQL Injection security check violations seen by the Application Firewall.

appFirewallViolXmlXssViolations (1.3.6.1.4.1.5951.4.1.1.64.1.21)

Number of XML Cross-Site Scripting (XSS) security check violations seen by the Application Firewall.

appFirewallViolXmlAttachmentViolations (1.3.6.1.4.1.5951.4.1.1.64.1.22)

Number of XML Attachment security check violations seen by the Application Firewall.

appFirewallViolCSRFtag (1.3.6.1.4.1.5951.4.1.1.64.1.23)

Number of Cross Site Request Forgery form tag security check violations seen by the Application Firewall.

appFirewallViolRefererHeader (1.3.6.1.4.1.5951.4.1.1.64.1.24)

Number of Referer Header security check violations seen by the Application Firewall.

appFirewallViolXmlSoapFaultViolations (1.3.6.1.4.1.5951.4.1.1.64.1.25)

Number of requests returning soap:fault from the backend server

appFirewallRet4xx (1.3.6.1.4.1.5951.4.1.1.64.1.26)

Number of requests returning HTTP 4xx from the backend server

appFirewallRet5xx (1.3.6.1.4.1.5951.4.1.1.64.1.27)

Number of requests returning HTTP 5xx from the backend server

appFirewallReqBytes (1.3.6.1.4.1.5951.4.1.1.64.1.28)

Number of bytes transferred for requests

appFirewallResBytes (1.3.6.1.4.1.5951.4.1.1.64.1.29)

Number of bytes transferred for responses

appFirewallLongAvgRespTime (1.3.6.1.4.1.5951.4.1.1.64.1.30)

Average backend response time in milliseconds since reboot

appFirewallShortAvgRespTime (1.3.6.1.4.1.5951.4.1.1.64.1.31)

Average backend response time in milliseconds over the last 7 seconds

appFirewallViolXmlGenViolations (1.3.6.1.4.1.5951.4.1.1.64.1.32)

Number of requests returning XML generic error from the backend server

appFirewallViolSignature (1.3.6.1.4.1.5951.4.1.1.64.1.33)

Number of Signature violations seen by the Application Firewall.

appFirewallTrapsDropped (1.3.6.1.4.1.5951.4.1.1.64.1.34)

AppFirewall SNMP traps dropped due to time limit.

appfwProfileTable (1.3.6.1.4.1.5951.4.1.1.64.2)

Indexed on: [appfwprofileName](#)

appfwprofileName (1.3.6.1.4.1.5951.4.1.1.64.2.1.1)

The name of the Application Firewall profile

appfwappFirewallRequestsPerProfile (1.3.6.1.4.1.5951.4.1.1.64.2.1.2)

HTTP/HTTPS requests sent to your protected web servers via the Application Firewall.

appfwappFirewallResponsesPerProfile (1.3.6.1.4.1.5951.4.1.1.64.2.1.3)

HTTP/HTTPS responses sent by your protected web servers via the Application Firewall.

appfwappFirewallAbortsPerProfile (1.3.6.1.4.1.5951.4.1.1.64.2.1.4)

Incomplete HTTP/HTTPS requests aborted by the client before the Application Firewall could finish processing them.

appfwappFirewallRedirectsPerProfile (1.3.6.1.4.1.5951.4.1.1.64.2.1.5)

HTTP/HTTPS requests redirected by the Application Firewall to a different Web page or web server. (HTTP 302)

appfwappFirewallViolStartURLPerProfile (1.3.6.1.4.1.5951.4.1.1.64.2.1.6)

Number of Start URL security check violations seen by the Application Firewall.

appfwappFirewallViolDenyURLPerProfile (1.3.6.1.4.1.5951.4.1.1.64.2.1.7)

Number of Deny URL security check violations seen by the Application Firewall.

appfwappFirewallViolRefererHeaderPerProfile (1.3.6.1.4.1.5951.4.1.1.64.2.1.8)

Number of Referer Header security check violations seen by the Application Firewall.

appfwappFirewallViolBufferOverflowPerProfile (1.3.6.1.4.1.5951.4.1.1.64.2.1.9)

Number of Buffer Overflow security check violations seen by the Application Firewall.

appfwappFirewallViolCSRFTagPerProfile (1.3.6.1.4.1.5951.4.1.1.64.2.1.10)

Number of Cross Site Request Forgery form tag security check violations seen by the Application Firewall.

appfwappFirewallViolCookiePerProfile (1.3.6.1.4.1.5951.4.1.1.64.2.1.11)

Number of Cookie Consistency security check violations seen by the Application Firewall.

appfwappFirewallViolXSSPerProfile (1.3.6.1.4.1.5951.4.1.1.64.2.1.12)

Number of HTML Cross-Site Scripting security check violations seen by the Application Firewall.

appfwappFirewallViolSQLPerProfile (1.3.6.1.4.1.5951.4.1.1.64.2.1.13)

Number of HTML SQL Injection security check violations seen by the Application Firewall.

appfwappFirewallViolFieldformatPerProfile (1.3.6.1.4.1.5951.4.1.1.64.2.1.14)

Number of Field Format security check violations seen by the Application Firewall.

appfwappFirewallViolFieldConsistencyPerProfile (1.3.6.1.4.1.5951.4.1.1.64.2.1.15)

Number of Field Consistency security check violations seen by the Application Firewall.

appfwappFirewallViolCreditCardPerProfile (1.3.6.1.4.1.5951.4.1.1.64.2.1.16)

Number of Credit Card security check violations seen by the Application Firewall.

appfwappFirewallViolSafeObjectPerProfile (1.3.6.1.4.1.5951.4.1.1.64.2.1.17)

Number of Safe Object security check violations seen by the Application Firewall.

appfwappFirewallViolWellformednessViolationsPerProfile (1.3.6.1.4.1.5951.4.1.1.64.2.1.18)

Number of XML Format security check violations seen by the Application Firewall.

appfwappFirewallViolXdosViolationsPerProfile (1.3.6.1.4.1.5951.4.1.1.64.2.1.19)

Number of XML Denial-of-Service security check violations seen by the Application Firewall.

appfwappFirewallViolMsgValViolationsPerProfile (1.3.6.1.4.1.5951.4.1.1.64.2.1.20)

Number of XML Message Validation security check violations seen by the Application Firewall.

appfwappFirewallViolWSIViolationsPerProfile (1.3.6.1.4.1.5951.4.1.1.64.2.1.21)

Number of Web Services Interoperability (WS-I) security check violations seen by the Application Firewall.

appfwappFirewallViolXmlSqlViolationsPerProfile (1.3.6.1.4.1.5951.4.1.1.64.2.1.22)

Number of XML SQL Injection security check violations seen by the Application Firewall.

appfwappFirewallViolXmlXssViolationsPerProfile (1.3.6.1.4.1.5951.4.1.1.64.2.1.23)

Number of XML Cross-Site Scripting (XSS) security check violations seen by the Application Firewall.

appfwappFirewallViolXmlAttachmentViolationsPerProfile (1.3.6.1.4.1.5951.4.1.1.64.2.1.24)

Number of XML Attachment security check violations seen by the Application Firewall.

appfwappFirewallTotalViolPerProfile (1.3.6.1.4.1.5951.4.1.1.64.2.1.25)

Number of violations seen by the application firewall on per profile basis

appfwappFirewallRet4xxPerProfile (1.3.6.1.4.1.5951.4.1.1.64.2.1.26)

Number of requests returning HTTP 4xx from the backend server

appfwappFirewallRet5xxPerProfile (1.3.6.1.4.1.5951.4.1.1.64.2.1.27)

Number of requests returning HTTP 5xx from the backend server

appfwappFirewallViolXmlSoapFaultViolationsPerProfile (1.3.6.1.4.1.5951.4.1.1.64.2.1.28)

Number of requests returning soap:fault from the backend server

appfwappFirewallReqBytesPerProfile (1.3.6.1.4.1.5951.4.1.1.64.2.1.29)

Number of bytes transferred for requests

appfwappFirewallResBytesPerProfile (1.3.6.1.4.1.5951.4.1.1.64.2.1.30)

Number of bytes transferred for responses

appfwappFirewallLongAvgRespTimePerProfile (1.3.6.1.4.1.5951.4.1.1.64.2.1.31)

Average backend response time in milliseconds since reboot

appfwappFirewallShortAvgRespTimePerProfile (1.3.6.1.4.1.5951.4.1.1.64.2.1.32)

Average backend response time in milliseconds over the last 7 seconds

appfwappFirewallViolXmlGenericViolationsPerProfile (1.3.6.1.4.1.5951.4.1.1.64.2.1.33)

Number of requests returning XML generic violation from the backend server

appfwappFirewallViolSignaturePerProfile (1.3.6.1.4.1.5951.4.1.1.64.2.1.34)

Number of Signature violations seen by the Application Firewall.

nsRnatGlobalStats (1.3.6.1.4.1.5951.4.1.1.65.1)

rnatTotRxBytes (1.3.6.1.4.1.5951.4.1.1.65.1.1)

Bytes received during RNAT sessions.

rnatTotTxBytes (1.3.6.1.4.1.5951.4.1.1.65.1.2)

Bytes sent during RNAT sessions.

rnatTotRxPkts (1.3.6.1.4.1.5951.4.1.1.65.1.3)

Packets received during RNAT sessions.

rnatTotTxPkts (1.3.6.1.4.1.5951.4.1.1.65.1.4)

Packets sent during RNAT sessions.

rnatTotTxSyn (1.3.6.1.4.1.5951.4.1.1.65.1.5)

Requests for connections sent during RNAT sessions.

rnatCurSessions (1.3.6.1.4.1.5951.4.1.1.65.1.6)

Currently active RNAT sessions.

nsRnatPerIPStatsTable (1.3.6.1.4.1.5951.4.1.1.65.2)

This table contains statistics related to rnat for the natip

Indexed on: [ipAddr](#)

ipRnatTotRxBytes (1.3.6.1.4.1.5951.4.1.1.65.2.1.1)

Bytes received on this IP address during RNAT sessions.

ipRnatTotTxBytes (1.3.6.1.4.1.5951.4.1.1.65.2.1.2)

Bytes sent from this IP address during RNAT sessions.

ipRnatTotRxPkts (1.3.6.1.4.1.5951.4.1.1.65.2.1.3)

Packets received on this IP address during RNAT sessions.

ipRnatTotTxPkts (1.3.6.1.4.1.5951.4.1.1.65.2.1.4)

Packets sent from this IP address during RNAT sessions.

ipRnatTotTxSyn (1.3.6.1.4.1.5951.4.1.1.65.2.1.5)

Requests for connections sent from this IP address during RNAT sessions.

ipRnatCurSessions (1.3.6.1.4.1.5951.4.1.1.65.2.1.6)

Currently active RNAT sessions started from this IP address.

piPolicyTable (1.3.6.1.4.1.5951.4.1.1.69.1)

The policy relationship table

Indexed on: [piPolName](#)

piPolName (1.3.6.1.4.1.5951.4.1.1.69.1.1.1)

Encoded name of the PI policy

piPolicyHits (1.3.6.1.4.1.5951.4.1.1.69.1.1.2)

Number of hits on the policy

piPolicyUndefHits (1.3.6.1.4.1.5951.4.1.1.69.1.1.3)

Number of undef hits on the policy

piPolFullName (1.3.6.1.4.1.5951.4.1.1.69.1.1.4)

Full name of the PI policy

nsInatGlobalStats (1.3.6.1.4.1.5951.4.1.1.74.1)**nat46TotTcp46 (1.3.6.1.4.1.5951.4.1.1.74.1.1)**

Total TCP packets translated (V4->v6).

nat46TotUdp46 (1.3.6.1.4.1.5951.4.1.1.74.1.2)

Total UDP packets translated (V4->v6).

nat46TotIcmp46 (1.3.6.1.4.1.5951.4.1.1.74.1.3)

Total ICMP packets translated (V4->v6).

nat46TotDrop46 (1.3.6.1.4.1.5951.4.1.1.74.1.4)

Total IPV4 packets dropped.

nat46TotTcp64 (1.3.6.1.4.1.5951.4.1.1.74.1.5)

Total TCP packets translated (V6->v4).

nat46TotUdp64 (1.3.6.1.4.1.5951.4.1.1.74.1.6)

Total UDP packets translated (V6->v4).

nat46TotIcmp64 (1.3.6.1.4.1.5951.4.1.1.74.1.7)

Total ICMP packets translated (V6->v4).

nat46TotDrop64 (1.3.6.1.4.1.5951.4.1.1.74.1.8)

Total IPV6 packets dropped.

nsInatPerNat46StatsTable (1.3.6.1.4.1.5951.4.1.1.74.2)

This provides statistics related to per nat46 rule

Indexed on: [inatname](#)

inatname (1.3.6.1.4.1.5951.4.1.1.74.2.1.1)

The name of the INAT

inatNat46Tcp46 (1.3.6.1.4.1.5951.4.1.1.74.2.1.2)

TCP packets translated (V4->v6).

inatNat46Udp46 (1.3.6.1.4.1.5951.4.1.1.74.2.1.3)

UDP packets translated (V4->v6).

inatNat46Icmp46 (1.3.6.1.4.1.5951.4.1.1.74.2.1.4)

ICMP packets translated (V4->v6).

inatNat46Drop46 (1.3.6.1.4.1.5951.4.1.1.74.2.1.5)

IPV4 packets dropped.

inatNat46Tcp64 (1.3.6.1.4.1.5951.4.1.1.74.2.1.6)

TCP packets translated (V6->v4).

inatNat46Udp64 (1.3.6.1.4.1.5951.4.1.1.74.2.1.7)

UDP packets translated (V6->v4).

inatNat46Icmp64 (1.3.6.1.4.1.5951.4.1.1.74.2.1.8)

ICMP packets translated (V6->v4).

inatNat46Drop64 (1.3.6.1.4.1.5951.4.1.1.74.2.1.9)

IPV6 packets dropped.

nsInatPerNatStatsTable (1.3.6.1.4.1.5951.4.1.1.74.3)

This provides statistics related to per inat session

Indexed on: [inat44name](#)

inat44name (1.3.6.1.4.1.5951.4.1.1.74.3.1.1)

The name of the INAT

inatTotHits (1.3.6.1.4.1.5951.4.1.1.74.3.1.2)

INAT total sessions

inatCurSessions (1.3.6.1.4.1.5951.4.1.1.74.3.1.3)

INAT current sessions

inatTotReceiveBytes (1.3.6.1.4.1.5951.4.1.1.74.3.1.4)

INAT total Received Bytes

inatTotSentBytes (1.3.6.1.4.1.5951.4.1.1.74.3.1.5)

INAT total Sent Bytes

inatTotpktreceived (1.3.6.1.4.1.5951.4.1.1.74.3.1.6)

INAT total Packets Received

inatTotpktsent (1.3.6.1.4.1.5951.4.1.1.74.3.1.7)

INAT total Packets Sent

nsNat64GlobalStats (1.3.6.1.4.1.5951.4.1.1.75.1)

nat64TotUdpSessions (1.3.6.1.4.1.5951.4.1.1.75.1.1)

Total number of UDP sessions created by NAT64.

nat64TotTcpSessions (1.3.6.1.4.1.5951.4.1.1.75.1.2)

Total number of TCP sessions created by NAT64.

nat64TotSessions (1.3.6.1.4.1.5951.4.1.1.75.1.3)

Total number of sessions created by NAT64.

nat64TotIcmpSessions (1.3.6.1.4.1.5951.4.1.1.75.1.4)

Total number of ICMP sessions created by NAT64.

nsLLDPConfigGroup (1.3.6.1.4.1.5951.4.1.1.80.1)

lldpMessageTxInterval (1.3.6.1.4.1.5951.4.1.1.80.1.1)

Time interval at which lldp packets will be sent.

lldpMessageTxHoldMultiplier (1.3.6.1.4.1.5951.4.1.1.80.1.2)

The time-to-live value expressed as a multiple of the lldpMessageTxInterval and lldpMessageTxHoldMultiplier.

nsLLDPStatsGroup (1.3.6.1.4.1.5951.4.1.1.80.2)

nsLLDPStatsTxPortTable (1.3.6.1.4.1.5951.4.1.1.80.2.1)

This table contains information about statistics of LLDP packets transmission.

Indexed on: [lldpStatsTxPortNum](#)

lldpStatsTxPortNum (1.3.6.1.4.1.5951.4.1.1.80.2.1.1.1)

Interface number

IldpStatsTxPortFramesTotal (1.3.6.1.4.1.5951.4.1.1.80.2.1.1.2)

Total LLDP Packets transmitted

nsLLDPStatsRxPortTable (1.3.6.1.4.1.5951.4.1.1.80.2.2)

This table contains information about statistics of recieved LLDP packets.

Indexed on: [IldpStatsRxPortNum](#)

IldpStatsRxPortNum (1.3.6.1.4.1.5951.4.1.1.80.2.2.1.1)

Interface number

IldpStatsRxPortTLVsDiscardedTotal (1.3.6.1.4.1.5951.4.1.1.80.2.2.1.2)

Total discarded LLDP packets.

IldpStatsRxPortFramesErrors (1.3.6.1.4.1.5951.4.1.1.80.2.2.1.3)

Total errors in LLDP packets.

IldpStatsRxPortFramesTotal (1.3.6.1.4.1.5951.4.1.1.80.2.2.1.4)

Total LLDP Packets received.

IldpStatsRxPortTLVsUnrecognizedTotal (1.3.6.1.4.1.5951.4.1.1.80.2.2.1.5)

Total TLVs not Recognised.

IldpStatsRemTablesLastChangeTime (1.3.6.1.4.1.5951.4.1.1.80.2.3)

Time stamp at which last change in Remote table seen.

IldpStatsRemTablesInserts (1.3.6.1.4.1.5951.4.1.1.80.2.4)

Total inserts in Remote LLDP table.

IldpStatsRemTablesDeletes (1.3.6.1.4.1.5951.4.1.1.80.2.5)

Total deletes in Remote LLDP table.

IldpStatsRemTablesAgeouts (1.3.6.1.4.1.5951.4.1.1.80.2.6)

Total Aged out entries in Remote LLDP table.

nsLLDPLocSystemsGroup (1.3.6.1.4.1.5951.4.1.1.80.3)

nsLLDPLocPortTable (1.3.6.1.4.1.5951.4.1.1.80.3.1)

This Table contains LLDP information of local port.

Indexed on: [IldpLocPortNum](#)

IldpLocPortNum (1.3.6.1.4.1.5951.4.1.1.80.3.1.1.1)

Interface Number

IldpLocPortIdSubtype (1.3.6.1.4.1.5951.4.1.1.80.3.1.1.2)

Local port id sub type

IldpLocPortId (1.3.6.1.4.1.5951.4.1.1.80.3.1.1.3)

Local port id

nsLLDPLocManAddrTable (1.3.6.1.4.1.5951.4.1.1.80.3.2)

This Table contains LLDP information of local management address.

Indexed on: [IldpLocManAddrSubtype](#)

IldpLocManAddrSubtype (1.3.6.1.4.1.5951.4.1.1.80.3.2.1.1)

Management address subtype of Remote System

IldpLocManAddr (1.3.6.1.4.1.5951.4.1.1.80.3.2.1.2)

Management address of Local System

IldpLocManAddrIfSubtype (1.3.6.1.4.1.5951.4.1.1.80.3.2.1.3)

Interface subtype

IldpLocManAddrIfId (1.3.6.1.4.1.5951.4.1.1.80.3.2.1.4)

interface id

IldpLocManAddrOID (1.3.6.1.4.1.5951.4.1.1.80.3.2.1.5)

Management address OID

IldpLocChassisIdSubtype (1.3.6.1.4.1.5951.4.1.1.80.3.3)

Local chassis Id type

IldpLocChassisId (1.3.6.1.4.1.5951.4.1.1.80.3.4)

Local chassis Id

IldpLocSysName (1.3.6.1.4.1.5951.4.1.1.80.3.5)

Local Sytem Name

IldpLocSysDesc (1.3.6.1.4.1.5951.4.1.1.80.3.6)

Sytem description

IldpLocSysCapSupported (1.3.6.1.4.1.5951.4.1.1.80.3.7)

Capabilities supported by local system

IldpLocSysCapEnabled (1.3.6.1.4.1.5951.4.1.1.80.3.8)

Capabilities enabled on local system

gslbSitesTable (1.3.6.1.4.1.5951.4.1.1.51.2.1)

This table contains gslb sites information

Indexed on: [siteName](#)

siteName (1.3.6.1.4.1.5951.4.1.1.51.2.1.1.1)

This is the name of the gslb site

sitelp (1.3.6.1.4.1.5951.4.1.1.51.2.1.1.2)

The private IP address of this GSLB site.

siteType (1.3.6.1.4.1.5951.4.1.1.51.2.1.1.3)

Indicates whether this GSLB site is local or remote.

siteMetricExchange (1.3.6.1.4.1.5951.4.1.1.51.2.1.1.4)

Indicates whether metric exchange is enabled or disabled at this GSLB site.

sitePublicIp (1.3.6.1.4.1.5951.4.1.1.51.2.1.1.6)

The public IP address of this GSLB site.

siteTotalRequests (1.3.6.1.4.1.5951.4.1.1.51.2.1.1.7)

Total number of requests received by the virtual servers represented by all GSLB services associated with this GSLB site.

siteTotalRequestBytes (1.3.6.1.4.1.5951.4.1.1.51.2.1.1.8)

Total number of request bytes received by the virtual servers represented by all GSLB services associated with this GSLB site.

siteTotalResponses (1.3.6.1.4.1.5951.4.1.1.51.2.1.1.9)

Number of responses received by the virtual servers represented by all GSLB services associated with this GSLB site.

siteTotalResponseBytes (1.3.6.1.4.1.5951.4.1.1.51.2.1.1.10)

Number of response bytes received by the virtual servers represented by all GSLB services associated with this GSLB site.

siteCurSrvrConnections (1.3.6.1.4.1.5951.4.1.1.51.2.1.1.11)

Number of current connections to the real servers behind the virtual servers represented by all GSLB services associated with this GSLB site.

siteCurCIntConnections (1.3.6.1.4.1.5951.4.1.1.51.2.1.1.12)

Number of current client connections to the virtual servers represented by all GSLB services associated with this GSLB site.

siteMetricMepStatus (1.3.6.1.4.1.5951.4.1.1.51.2.1.1.13)

Indicates the status of the site metric Metric Exchange connection at this GSLB site.

nwMetricMepStatus (1.3.6.1.4.1.5951.4.1.1.51.2.1.1.14)

Indicates the status of the network metric Metric Exchange connection at this GSLB site.

nwMetricExchange (1.3.6.1.4.1.5951.4.1.1.51.2.1.1.15)

Indicates whether network metric exchange is enabled or disabled at this GSLB site.

persExchange (1.3.6.1.4.1.5951.4.1.1.51.2.1.1.16)

Indicates whether Persistence entries exchange is enabled or disabled at this GSLB site.

gslbSiteInetAddressType (1.3.6.1.4.1.5951.4.1.1.51.2.1.1.17)

The address type of gslbSiteInetAddress

gslbSiteInetAddress (1.3.6.1.4.1.5951.4.1.1.51.2.1.1.18)

The internet address of the gslb site.

gslbSitePublicInetAddressType (1.3.6.1.4.1.5951.4.1.1.51.2.1.1.19)

The address type of gslbSitePublicInetAddress

gslbSitePublicInetAddress (1.3.6.1.4.1.5951.4.1.1.51.2.1.1.20)

The internet address of the gslb site public IP.

gslbPoliciesTable (1.3.6.1.4.1.5951.4.1.1.51.2.2)

This table contains the policy information

Indexed on: [gslbPolicyName](#)

gslbPolicyName (1.3.6.1.4.1.5951.4.1.1.51.2.2.1)

This is the policy name

totalHits (1.3.6.1.4.1.5951.4.1.1.51.2.2.1.2)

Total number of hits on this GSLB policy.

nsDomainTable (1.3.6.1.4.1.5951.4.1.1.51.3.1)

This table contains information about the Hits on the Domains.

Indexed on: [domainName](#)

domainName (1.3.6.1.4.1.5951.4.1.1.51.3.1.1.1)

The domain name

dnsTotalQueries (1.3.6.1.4.1.5951.4.1.1.51.3.1.1.2)

Total number of DNS queries received.

domainNameFull (1.3.6.1.4.1.5951.4.1.1.51.3.1.1.3)

Full Domain name string

scPolicyConfigTable (1.3.6.1.4.1.5951.4.1.1.55.2.1)

The sure connect policy configuration table

Indexed on: [scPolicyName](#)

scPolicyName (1.3.6.1.4.1.5951.4.1.1.55.2.1.1.1)

The name of Sure Connect policy.

scPolUrl (1.3.6.1.4.1.5951.4.1.1.55.2.1.1.2)

The URL in the IOH Policy.

scDelayThreshold (1.3.6.1.4.1.5951.4.1.1.55.2.1.1.3)

The delay threshold for sc policy.

scMaxConnections (1.3.6.1.4.1.5951.4.1.1.55.2.1.1.4)

The max connections for sc policy.

scActionType (1.3.6.1.4.1.5951.4.1.1.55.2.1.1.5)

The type of action that NetScaler takes when initiating on-hold.

scAlternateContentServiceName (1.3.6.1.4.1.5951.4.1.1.55.2.1.1.6)

The alternate service name for the content.

scRuleName (1.3.6.1.4.1.5951.4.1.1.55.2.1.1.7)

The rule that the NetScaler matches with the incoming request.

scAlternateContentPath (1.3.6.1.4.1.5951.4.1.1.55.2.1.1.8)

The alternate path for the content.

nsLLDPRemTable (1.3.6.1.4.1.5951.4.1.1.80.4.1)

This table contains lldp information of neighbors.

Indexed on: [lldpRemLocalPortNum](#)

lldpRemTimeMark (1.3.6.1.4.1.5951.4.1.1.80.4.1.1.1)

Time mark when lldp info recieved

lldpRemLocalPortNum (1.3.6.1.4.1.5951.4.1.1.80.4.1.1.2)

local nic no

IldpRemChassisIdSubtype (1.3.6.1.4.1.5951.4.1.1.80.4.1.1.3)

ChassisId subtype of remote system

IldpRemChassisId (1.3.6.1.4.1.5951.4.1.1.80.4.1.1.4)

Chassiss id of remote system

IldpRemPortIdSubtype (1.3.6.1.4.1.5951.4.1.1.80.4.1.1.5)

port id subtype of remote system

IldpRemPortId (1.3.6.1.4.1.5951.4.1.1.80.4.1.1.6)

port id of remote system

IldpRemPortDesc (1.3.6.1.4.1.5951.4.1.1.80.4.1.1.7)

port description of remote system

IldpRemSysName (1.3.6.1.4.1.5951.4.1.1.80.4.1.1.8)

remote system name

IldpRemSysDesc (1.3.6.1.4.1.5951.4.1.1.80.4.1.1.9)

remote system description

IldpRemSysCapSupported (1.3.6.1.4.1.5951.4.1.1.80.4.1.1.10)

remote system capabilities

IldpRemSysCapEnabled (1.3.6.1.4.1.5951.4.1.1.80.4.1.1.11)

enabled capabilities of system

nsLLDPRemManAddrTable (1.3.6.1.4.1.5951.4.1.1.80.4.2)

This table contains LLDP information about management address of neighbors.

Indexed on: [IldpRemLocalPortNum](#), [IldpRemManAddr](#)

IldpRemManAddrSubtype (1.3.6.1.4.1.5951.4.1.1.80.4.2.1.1)

Management address subtype of Remote System

IldpRemManAddr (1.3.6.1.4.1.5951.4.1.1.80.4.2.1.2)

Management address of Remote System

IldpRemManAddrIfSubtype (1.3.6.1.4.1.5951.4.1.1.80.4.2.1.3)

Interface subtype

IldpRemManAddrIfId (1.3.6.1.4.1.5951.4.1.1.80.4.2.1.4)

interface id

IldpRemManAddrOID (1.3.6.1.4.1.5951.4.1.1.80.4.2.1.5)

Management address OID

serviceTable (1.3.6.1.4.1.5951.4.1.2.1)

The netscaler services table

Indexed on: [svcServiceName](#)

svcServiceName (1.3.6.1.4.1.5951.4.1.2.1.1.1)

The name of the service.

svcIpAddress (1.3.6.1.4.1.5951.4.1.2.1.1.2)

The ip address at which the service is running.

svcPort (1.3.6.1.4.1.5951.4.1.2.1.1.3)

The port at which the service is running.

svcServiceType (1.3.6.1.4.1.5951.4.1.2.1.1.4)

The protocol type of the service

svcState (1.3.6.1.4.1.5951.4.1.2.1.1.5)

Current state of the server. Possible values are UP, DOWN, UNKNOWN, OFS(Out of Service), TROFS(Transition Out of Service), TROFS_DOWN(Down When going Out of Service)

svcMaxReqPerConn (1.3.6.1.4.1.5951.4.1.2.1.1.6)

Maximum requests per connection allowed on this service.

svcAvgTransactionTime (1.3.6.1.4.1.5951.4.1.2.1.1.7)

Average transaction time in microseconds between netscaler and the service behind it.

svcEstablishedConn (1.3.6.1.4.1.5951.4.1.2.1.1.8)

Total number of connections in ESTABLISHED state.

svcActiveConn (1.3.6.1.4.1.5951.4.1.2.1.1.9)

Number of connections that are currently active.

svcSurgeCount (1.3.6.1.4.1.5951.4.1.2.1.1.10)

Number of requests in the surge queue.

svcTotalRequests (1.3.6.1.4.1.5951.4.1.2.1.1.30)

Total number of requests received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

svcTotalRequestBytes (1.3.6.1.4.1.5951.4.1.2.1.1.31)

Total number of request bytes received on this service or virtual server.

svcTotalResponses (1.3.6.1.4.1.5951.4.1.2.1.1.32)

Number of responses received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

svcTotalResponseBytes (1.3.6.1.4.1.5951.4.1.2.1.1.33)

Number of response bytes received by this service or virtual server.

svcTotalPktsRecvd (1.3.6.1.4.1.5951.4.1.2.1.1.34)

Total number of packets received by this service or virtual server.

svcTotalPktsSent (1.3.6.1.4.1.5951.4.1.2.1.1.35)

Total number of packets sent.

svcTotalSynsRecvd (1.3.6.1.4.1.5951.4.1.2.1.1.36)

Total number of SYN packets received from clients on this service (only when directly accessed) or virtual server.

svcGslbSiteName (1.3.6.1.4.1.5951.4.1.2.1.1.37)

The name of the gslb site on which this service is defined.

svcAvgSvrTTFB (1.3.6.1.4.1.5951.4.1.2.1.1.38)

Average TTFB between the NetScaler appliance and the server. TTFB is the time interval between sending the request packet to a service and receiving the first response from the service

svctotalJsTransactions (1.3.6.1.4.1.5951.4.1.2.1.1.39)

Total number of javascripts sent to genuine clients.

svcdosQDepth (1.3.6.1.4.1.5951.4.1.2.1.1.40)

Number of clients waiting currently in priority queue

svcCurCIntConnections (1.3.6.1.4.1.5951.4.1.2.1.1.41)

Number of current client connections.

svcRequestRate (1.3.6.1.4.1.5951.4.1.2.1.1.42)

Request rate in requests per second for this service or virtual server.

svcRxBytesRate (1.3.6.1.4.1.5951.4.1.2.1.1.43)

Request rate in bytes per second for this service or virtual server.

svcTxBytesRate (1.3.6.1.4.1.5951.4.1.2.1.1.44)

Response rate in bytes per second for this service or virtual server.

svcSynfloodRate (1.3.6.1.4.1.5951.4.1.2.1.1.45)

Rate of unacknowledged SYN packets for this service or virtual server.

svcTicksSinceLastStateChange (1.3.6.1.4.1.5951.4.1.2.1.1.47)

Time (in 10 milliseconds) since the last state change.

svcTotalClients (1.3.6.1.4.1.5951.4.1.2.1.1.48)

Total number of established client connections.

svcTotalServers (1.3.6.1.4.1.5951.4.1.2.1.1.49)

Total number of established server connections.

svcMaxClients (1.3.6.1.4.1.5951.4.1.2.1.1.52)

Maximum open connections allowed on this service.

svcActiveTransactions (1.3.6.1.4.1.5951.4.1.2.1.1.53)

Number of active transactions handled by this service. (Including those in the surge queue.)

Active Transaction means number of transactions currently served by the server including those waiting in the SurgeQ

svcServiceFullName (1.3.6.1.4.1.5951.4.1.2.1.1.54)

The name of the service.

svclnetAddressType (1.3.6.1.4.1.5951.4.1.2.1.1.55)

The address type of svclnetAddress

svclnetAddress (1.3.6.1.4.1.5951.4.1.2.1.1.56)

The Internet address at which the service is running.

svcTidId (1.3.6.1.4.1.5951.4.1.2.1.1.57)

Traffic Domain ID of this service.

svcGslbState (1.3.6.1.4.1.5951.4.1.2.1.1.58)

Effective state of the gslb service. Possible values are UP, DOWN, UNKNOWN, OFS(Out of Service), TROFS (Transition Out of Service), TROFS_DOWN(Down When going Out of Service)

serverTable (1.3.6.1.4.1.5951.4.1.2.2)

The servers table.

Indexed on: [serverName](#)

serverName (1.3.6.1.4.1.5951.4.1.2.2.1.1)

The name of the server.

serverIpAddress (1.3.6.1.4.1.5951.4.1.2.2.1.2)

The IP address of the server.

serverState (1.3.6.1.4.1.5951.4.1.2.2.1.3)

The state of the server.

serverDelay (1.3.6.1.4.1.5951.4.1.2.2.1.4)

Specifies time in seconds after which all services in this server are brought down.

serverFullName (1.3.6.1.4.1.5951.4.1.2.2.1.5)

The name of the server.

serverInetAddressType (1.3.6.1.4.1.5951.4.1.2.2.1.6)

The address type of serverInetAddress

serverInetAddress (1.3.6.1.4.1.5951.4.1.2.2.1.7)

The Internet address of the server.

serverTldId (1.3.6.1.4.1.5951.4.1.2.2.1.8)

Traffic Domain ID of this server.

serviceScpolicyTable (1.3.6.1.4.1.5951.4.1.2.3)

The service sure connect policy relationship table

Indexed on: [svcServiceName](#), [scPolicyName](#)

svcscpolicyPrimaryIPAddress (1.3.6.1.4.1.5951.4.1.2.3.1.3)

The IP address of the service or virtual server to which the policy is bound.

svcscpolicyPrimaryPort (1.3.6.1.4.1.5951.4.1.2.3.1.4)

The port of the service or virtual server to which the policy is bound.

svcscpolicydesIpAddress (1.3.6.1.4.1.5951.4.1.2.3.1.10)

IP address of the destination service.

svcscpolicydestPort (1.3.6.1.4.1.5951.4.1.2.3.1.11)

Port number of the destination service.

svcscpolicyavgServerTransactionTime (1.3.6.1.4.1.5951.4.1.2.3.1.12)

Average server transaction time in seconds for this SureConnect Policy.

svcscpolicytotClientTransaction (1.3.6.1.4.1.5951.4.1.2.3.1.13)

Total number of client transactions processed by this SureConnect policy.

svcscpolicytotOpenConn (1.3.6.1.4.1.5951.4.1.2.3.1.14)

Current number of open connections to the servers matching this policy.

svcscpolicyscPhysicalServiceIP (1.3.6.1.4.1.5951.4.1.2.3.1.15)

IP address of the service for which these statistics are maintained.

svcscpolicyscPhysicalServicePort (1.3.6.1.4.1.5951.4.1.2.3.1.16)

Port of the service for which these statistics are maintained.

svcscpolicyscCurrentWaitingTime (1.3.6.1.4.1.5951.4.1.2.3.1.17)

Value of the currently estimated waiting time in seconds for the configured URL.

svcscpolicyscCurrentClientConnections (1.3.6.1.4.1.5951.4.1.2.3.1.18)

Number of clients currently allowed a server connection by this SureConnect policy.

svcscpolicyscTotalClientConnections (1.3.6.1.4.1.5951.4.1.2.3.1.19)

Total number of clients that were allowed a server connection by this SureConnect policy.

svcscpolicyscTotalServerConnections (1.3.6.1.4.1.5951.4.1.2.3.1.20)

Total number of server connections that were established through this SureConnect policy.

svcscpolicyscTotalRequestsReceived (1.3.6.1.4.1.5951.4.1.2.3.1.21)

Total number of requests received by this SureConnect policy.

svcscpolicyscTotalRequestBytes (1.3.6.1.4.1.5951.4.1.2.3.1.22)

Total number of request bytes received by this SureConnect policy.

svcscpolicyscTotalResponsesReceived (1.3.6.1.4.1.5951.4.1.2.3.1.23)

Total number of server responses received by this SureConnect policy.

svcscpolicyscTotalResponseBytes (1.3.6.1.4.1.5951.4.1.2.3.1.24)

Total number of response bytes received by this SureConnect policy.

svcscpolicyscCurrentSurgeQClients (1.3.6.1.4.1.5951.4.1.2.3.1.25)

Number of clients currently matching the SureConnect policy, but are in the surge queue.

svcscpolicyscCurrentWaitingClients (1.3.6.1.4.1.5951.4.1.2.3.1.26)

Current number of SureConnect priority clients that are waiting for a server connection.

svcscpolicyscTotalServerTransactions (1.3.6.1.4.1.5951.4.1.2.3.1.27)

Number of 200 OK responses received from the web server by this SureConnect policy.

svcscpolicyscTotalServerTTFBTransactions (1.3.6.1.4.1.5951.4.1.2.3.1.28)

Number of Time-To-First-Byte transactions from the web server for this SureConnect policy.

svcscpolicyscTotalServerTTLB (1.3.6.1.4.1.5951.4.1.2.3.1.29)

Server Time-To-Last-Byte in seconds calculated for this SureConnect policy.

svcscpolicyscTotalClientTTLB (1.3.6.1.4.1.5951.4.1.2.3.1.30)

Client Time-To-Last-Byte in seconds calculated for this SureConnect policy.

svcscpolycscTotalServerTTFB (1.3.6.1.4.1.5951.4.1.2.3.1.31)

Server Time-To-First-Byte in seconds calculated for this SureConnect policy.

svcscpolycscAverageClientTTLB (1.3.6.1.4.1.5951.4.1.2.3.1.32)

Average value of the client Time-To-Last-Byte in seconds for this SureConnect policy.

svcscpolycscAverageServerTTFB (1.3.6.1.4.1.5951.4.1.2.3.1.33)

Average value of the server Time-To-First-Byte in seconds for this SureConnect policy.

serviceAdvanceSslConfigTable (1.3.6.1.4.1.5951.4.1.2.4)

The service advance SSL configuration

Indexed on: **svcServiceName**

svcSslIDH (1.3.6.1.4.1.5951.4.1.2.4.1.1)

Whether DH is enabled/disabled.

svcSslIDHCount (1.3.6.1.4.1.5951.4.1.2.4.1.2)

The DH refresh count to re-generate public/private key.

svcSslIDHFilePath (1.3.6.1.4.1.5951.4.1.2.4.1.3)

The DH file path name.

svcSslleRSA (1.3.6.1.4.1.5951.4.1.2.4.1.4)

The ephemeral RSA support for service.

svcSslleRSACount (1.3.6.1.4.1.5951.4.1.2.4.1.5)

The eRSA refresh count to re-generate RSA temporary key.

svcSslv2Protocol (1.3.6.1.4.1.5951.4.1.2.4.1.6)

The support for SSLv2 protocol for service.

svcSslv3Protocol (1.3.6.1.4.1.5951.4.1.2.4.1.7)

The support for SSLv3 protocol for service.

svcSslTLSv1Protocol (1.3.6.1.4.1.5951.4.1.2.4.1.8)

The support for TLSv1 protocol for service.

svcSslRedirectSupport (1.3.6.1.4.1.5951.4.1.2.4.1.9)

The support for ssl redirect for service.

svcSslClearTextPort (1.3.6.1.4.1.5951.4.1.2.4.1.10)

The clear text port on the backend webserver.

serviceCipherBindingTable (1.3.6.1.4.1.5951.4.1.2.5)

The service cipher bindings

Indexed on: **svcServiceName**, **svcSslCipherBindName**

svcSslCipherBindName (1.3.6.1.4.1.5951.4.1.2.5.1.1)

The cipher name bound to this service.

svcSslCipherBindDesc (1.3.6.1.4.1.5951.4.1.2.5.1.2)

The Cipher description.

serviceGlobalStatsGroup (1.3.6.1.4.1.5951.4.1.2.6)

svcCount (1.3.6.1.4.1.5951.4.1.2.6.1)

Number of services defined on this NetScaler appliance.

serverCount (1.3.6.1.4.1.5951.4.1.2.6.2)

Number of servers defined on this NetScaler appliance.

svccgroupCount (1.3.6.1.4.1.5951.4.1.2.6.3)

Number of service groups defined on this NetScaler appliance.

svccgroupmemCount (1.3.6.1.4.1.5951.4.1.2.6.4)

Number of service group members defined on this NetScaler appliance.

syssvcCount (1.3.6.1.4.1.5951.4.1.2.6.5)

Number of services configured on this NetScaler appliance.

sysupsvcCount (1.3.6.1.4.1.5951.4.1.2.6.6)

Number of configured services which are up on this NetScaler appliance.

sysupsvctmCount (1.3.6.1.4.1.5951.4.1.2.6.7)

Number of configured service items which are up on this NetScaler appliance.

serviceGroupMemberTable (1.3.6.1.4.1.5951.4.1.2.7)

The service group member bindings

Indexed on: [svcGrpMemberGroupName](#), [svcGrpMemberName](#)

svcGrpMemberGroupName (1.3.6.1.4.1.5951.4.1.2.7.1.1)

The name of the service Group

svcGrpMemberName (1.3.6.1.4.1.5951.4.1.2.7.1.2)

The name of the service group member

svcGrpMemberPrimaryIPAddress (1.3.6.1.4.1.5951.4.1.2.7.1.3)

The IP address on which the service is running.

svcGrpMemberPrimaryPort (1.3.6.1.4.1.5951.4.1.2.7.1.4)

The port on which the service is running.

svcGrpMemberServiceType (1.3.6.1.4.1.5951.4.1.2.7.1.5)

The service type of this service. Possible values are ADNS, DNS, MYSQL, RTSP, SSL_DIAMETER, ADNS_TCP, DNS_TCP, NNTP, SIP_UDP, SSL_TCP, ANY, FTP, RADIUS, SNMP, TCP, DHCPRA, HTTP, RDP, SSL, TFTP, DIAMETER, MSSQL, RPCSVR, SSL_BRIDGE, UDP

svcGrpMemberState (1.3.6.1.4.1.5951.4.1.2.7.1.6)

Current state of the server. Possible values are UP, DOWN, UNKNOWN, OFS(Out of Service), TROFS(Transition Out of Service), TROFS_DOWN(Down When going Out of Service)

svcGrpMemberWeight (1.3.6.1.4.1.5951.4.1.2.7.1.7)

The weight of the service group member tied to the vserver.

svcGrpMemberMaxReqPerConn (1.3.6.1.4.1.5951.4.1.2.7.1.8)

Maximum requests per connection allowed on this service.

svcGrpMemberAvgTransactionTime (1.3.6.1.4.1.5951.4.1.2.7.1.9)

Average transaction time in microseconds between netscaler and the service behind it.

svcGrpMemberEstablishedConn (1.3.6.1.4.1.5951.4.1.2.7.1.10)

Total number of connections in ESTABLISHED state.

svcGrpMemberActiveConn (1.3.6.1.4.1.5951.4.1.2.7.1.11)

Number of connections that are currently active.

svcGrpMemberSurgeCount (1.3.6.1.4.1.5951.4.1.2.7.1.12)

Number of requests in the surge queue.

svcGrpMemberTotalRequests (1.3.6.1.4.1.5951.4.1.2.7.1.13)

Total number of requests received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

svcGrpMemberTotalRequestBytes (1.3.6.1.4.1.5951.4.1.2.7.1.14)

Total number of request bytes received on this service or virtual server.

svcGrpMemberTotalResponses (1.3.6.1.4.1.5951.4.1.2.7.1.15)

Number of responses received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

svcGrpMemberTotalResponseBytes (1.3.6.1.4.1.5951.4.1.2.7.1.16)

Number of response bytes received by this service or virtual server.

svcGrpMemberTotalPktsRecvd (1.3.6.1.4.1.5951.4.1.2.7.1.17)

Total number of packets received by this service or virtual server.

svcGrpMemberTotalPktsSent (1.3.6.1.4.1.5951.4.1.2.7.1.18)

Total number of packets sent.

svcGrpMemberTotalSynsRecvd (1.3.6.1.4.1.5951.4.1.2.7.1.19)

Total number of SYN packets received from clients on this service (only when directly accessed) or virtual server.

svcGrpMemberGslbSiteName (1.3.6.1.4.1.5951.4.1.2.7.1.20)

The name of the gslb site on which this service is defined.

svcGrpMemberAvgSvrTTFB (1.3.6.1.4.1.5951.4.1.2.7.1.21)

Average TTFB between the NetScaler appliance and the server. TTFB is the time interval between sending the request packet to a service and receiving the first response from the service

svcGrpMembertotalJsTransactions (1.3.6.1.4.1.5951.4.1.2.7.1.22)

Total number of javascripts sent to genuine clients.

svcGrpMemberdosQDepth (1.3.6.1.4.1.5951.4.1.2.7.1.23)

Number of clients waiting currently in priority queue

svcGrpMemberCurCIntConnections (1.3.6.1.4.1.5951.4.1.2.7.1.24)

Number of current client connections.

svcGrpMemberRequestRate (1.3.6.1.4.1.5951.4.1.2.7.1.25)

Request rate in requests per second for this service or virtual server.

svcGrpMemberRxBytesRate (1.3.6.1.4.1.5951.4.1.2.7.1.26)

Request rate in bytes per second for this service or virtual server.

svcGrpMemberTxBytesRate (1.3.6.1.4.1.5951.4.1.2.7.1.27)

Response rate in bytes per second for this service or virtual server.

svcGrpMemberSynfloodRate (1.3.6.1.4.1.5951.4.1.2.7.1.28)

Rate of unacknowledged SYN packets for this service or virtual server.

svcGrpMemberTicksSinceLastStateChange (1.3.6.1.4.1.5951.4.1.2.7.1.31)

Time (in 10 milliseconds) since the last state change.

svcGrpMemberGroupFullName (1.3.6.1.4.1.5951.4.1.2.7.1.32)

The name of the service Group

svcGrpMemberFullName (1.3.6.1.4.1.5951.4.1.2.7.1.33)

The name of the service group member

svcGrpMemberPrimaryInetAddressType (1.3.6.1.4.1.5951.4.1.2.7.1.34)

The address type of svcGrpMemberPrimaryInetAddress

svcGrpMemberPrimaryInetAddress (1.3.6.1.4.1.5951.4.1.2.7.1.35)

The Internet address at which the service is running.

svcGrpMemberServerName (1.3.6.1.4.1.5951.4.1.2.7.1.36)

The name of the server of the servicegroup member

svcGrpMemberTidId (1.3.6.1.4.1.5951.4.1.2.7.1.37)

Traffic Domain ID of this service group member.

serviceDospolicyTable (1.3.6.1.4.1.5951.4.1.2.8)

The service DOS policy relationship table

Indexed on: [svcServiceName](#), [dosPolicyName](#)

svcdospolicydosTotJSSent (1.3.6.1.4.1.5951.4.1.2.8.1.1)

Total number of DoS JavaScript transactions performed for this policy.

svcdospolicydosTotJSBytesSent (1.3.6.1.4.1.5951.4.1.2.8.1.2)

Total number of DoS JavaScript bytes sent for this policy.

svcdospolicydosTotJSRefused (1.3.6.1.4.1.5951.4.1.2.8.1.3)

Number of times the DoS JavaScript was not sent because the set JavaScript rate was not met for this policy.

svcdospolicydosTotNonGetPostRequests (1.3.6.1.4.1.5951.4.1.2.8.1.4)

Number of non-GET and non-POST requests for which DOS JavaScript was sent.

svcdospolicydosPhysicalServiceIP (1.3.6.1.4.1.5951.4.1.2.8.1.5)

IP address of the service to which this policy is bound.

svcdospolicydosPhysicalServicePort (1.3.6.1.4.1.5951.4.1.2.8.1.6)

Port address of the service to which this policy is bound.

svcdospolicydosCurrentQueueSize (1.3.6.1.4.1.5951.4.1.2.8.1.7)

Current queue size of the server to which this policy is bound.

svcdospolicydosCurrentJSRate (1.3.6.1.4.1.5951.4.1.2.8.1.8)

Current rate at which JavaScript is being sent in response to client requests.

svcdospolicydosTotValidClients (1.3.6.1.4.1.5951.4.1.2.8.1.9)

Total number of valid DoS cookies received for this policy.

svcdospolicydosCurServerRespRate (1.3.6.1.4.1.5951.4.1.2.8.1.10)

Current rate at which the server to which this policy is bound is responding.

monitorMemberTable (1.3.6.1.4.1.5951.4.1.2.9)

The monitor table

Indexed on: [monitorName](#)

monitorName (1.3.6.1.4.1.5951.4.1.2.9.1.1)

Monitor name

responseTimeoutThreshold (1.3.6.1.4.1.5951.4.1.2.9.1.2)

Monitor Response timeout threshold, above which snmp trap will be fired.It is expressed in milliseconds.

monitorType (1.3.6.1.4.1.5951.4.1.2.9.1.3)

Type of the monitor.

monitorInterval (1.3.6.1.4.1.5951.4.1.2.9.1.4)

Interval between monitoring probes.It is expressed in milliseconds.

monitorResponseTimeout (1.3.6.1.4.1.5951.4.1.2.9.1.5)

Maximum time a monitor probe can take to respond.It is expressed in milliseconds.

monitorDowntime (1.3.6.1.4.1.5951.4.1.2.9.1.6)

Time for which the monitor probes are not fired once it is down.It is expressed in milliseconds.

monitorRetrys (1.3.6.1.4.1.5951.4.1.2.9.1.7)

Number of failed attempts to make server DOWN.

destinationIP (1.3.6.1.4.1.5951.4.1.2.9.1.8)

Destination IP address that is used for monitoring.

destinationPort (1.3.6.1.4.1.5951.4.1.2.9.1.9)

Destination port that is used for monitoring.

drtmDeviation (1.3.6.1.4.1.5951.4.1.2.9.1.10)

Tolerable Deviation of response time for DRTM.It is expressed in milliseconds.

drtmActiveMonitors (1.3.6.1.4.1.5951.4.1.2.9.1.11)

Number of monitors contributing to DRTM average.

drtmCumResponseTimeout (1.3.6.1.4.1.5951.4.1.2.9.1.12)

Total cumulative response time of all active DRTM monitors.It is expressed in milliseconds.

alarmProbeFailedRetries (1.3.6.1.4.1.5951.4.1.2.9.1.13)

Number of failed attempts to generate snmp trap.

destinationInetAddressType (1.3.6.1.4.1.5951.4.1.2.9.1.14)

The address type of destinationInetAddress

destinationInetAddress (1.3.6.1.4.1.5951.4.1.2.9.1.15)

Destination Internet address that is used for monitoring.

monServiceMemberTable (1.3.6.1.4.1.5951.4.1.2.10)

The moninfo table, bindings of monitors to services.

Indexed on: [monServiceName](#), [monitorName](#)

monServiceName (1.3.6.1.4.1.5951.4.1.2.10.1.1)

The name of the service to which the monitor is bound.

monitorRTO (1.3.6.1.4.1.5951.4.1.2.10.1.2)

Response time in micro-seconds. (Calculated using LRTM.)

monitorState (1.3.6.1.4.1.5951.4.1.2.10.1.3)

State of the specified monitor. Possible states are UP, OUT OF SERVICE, DOWN, GOING OUT OF SERVICE, and DOWN WHEN GOING OUT OF SERVICE.

drtmRTO (1.3.6.1.4.1.5951.4.1.2.10.1.4)

Monitor probe time in milli-seconds for DRTM monitors. (Round trip time)

drtmLearningProbes (1.3.6.1.4.1.5951.4.1.2.10.1.5)

Current number of pending DRTM monitoring probes.

monitorCurFailedCount (1.3.6.1.4.1.5951.4.1.2.10.1.6)

Current, continuous monitoring probe failure count. (Reset on success only.)

monitorWeight (1.3.6.1.4.1.5951.4.1.2.10.1.7)

Weight assigned to the monitor binding.

alarmMonrespto (1.3.6.1.4.1.5951.4.1.2.10.1.8)

This is the response time taken for the current monitor probe.

monitorProbes (1.3.6.1.4.1.5951.4.1.2.10.1.9)

Number of monitoring probes sent.

monitorFailed (1.3.6.1.4.1.5951.4.1.2.10.1.10)

Number of failed monitoring probes.

monitorMaxClient (1.3.6.1.4.1.5951.4.1.2.10.1.11)

Number of monitoring probes that were not sent due to MaxClients.

monitorFailedCon (1.3.6.1.4.1.5951.4.1.2.10.1.12)

Number of failed monitoring probes due to failed connections.

monitorFailedCode (1.3.6.1.4.1.5951.4.1.2.10.1.13)

Number of failed monitoring probes due to improper response code.

monitorFailedStr (1.3.6.1.4.1.5951.4.1.2.10.1.14)

Number of failed monitoring probes due to invalid response string.

monitorFailedTimeout (1.3.6.1.4.1.5951.4.1.2.10.1.15)

Number of failed monitoring probes due to timeout.

monitorFailedSend (1.3.6.1.4.1.5951.4.1.2.10.1.16)

Number of failed monitoring probes due to inability to send the data.

monitorFailedFTP (1.3.6.1.4.1.5951.4.1.2.10.1.17)

Number of failed monitoring probes due to ftp protocol violation.

monitorFailedPort (1.3.6.1.4.1.5951.4.1.2.10.1.18)

Number of failed monitoring probes due to port unreachable response.

monitorFailedResponse (1.3.6.1.4.1.5951.4.1.2.10.1.19)

Number of failed monitoring probes due to invalid response.

monitorFailedId (1.3.6.1.4.1.5951.4.1.2.10.1.20)

Number of failed monitoring probes due to response id mismatch.

monitorProbesNoChange (1.3.6.1.4.1.5951.4.1.2.10.1.21)

Number of monitoring probes which did not change the state.

monitorResponseTimeoutThreshExceed (1.3.6.1.4.1.5951.4.1.2.10.1.22)

Number of times the response time has exceeded the configured threshold.

serviceGroupTable (1.3.6.1.4.1.5951.4.1.2.11)

The netscaler services group table

Indexed on: [svcgrpSvcGroupName](#)

svcgrpSvcGroupName (1.3.6.1.4.1.5951.4.1.2.11.1.1)

The name of the service Group

svcgrpSvcGroupType (1.3.6.1.4.1.5951.4.1.2.11.1.2)

The type of the service Group.

svcgrpSvcGroupState (1.3.6.1.4.1.5951.4.1.2.11.1.3)

The state of the service Group

svcgrpSvcGroupFullName (1.3.6.1.4.1.5951.4.1.2.11.1.4)

The full name of the service Group

svcgrpTdId (1.3.6.1.4.1.5951.4.1.2.11.1.5)

Traffic Domain ID of this service group.

vserverTable (1.3.6.1.4.1.5951.4.1.3.1)

The vservers table

Indexed on: [vsvrName](#)

vsvrName (1.3.6.1.4.1.5951.4.1.3.1.1.1)

The name of the vserver

vsvrIpAddress (1.3.6.1.4.1.5951.4.1.3.1.1.2)

IP address of the vserver

vsvrPort (1.3.6.1.4.1.5951.4.1.3.1.1.3)

the port of the vserver

vsvrType (1.3.6.1.4.1.5951.4.1.3.1.1.4)

Protocol associated with the vserver

vsvrState (1.3.6.1.4.1.5951.4.1.3.1.1.5)

Current state of the server. Possible values are UP, DOWN, UNKNOWN, OFS(Out of Service), TROFS(Transition Out of Service), TROFS_DOWN(Down When going Out of Service)

vsvrCurCIntConnections (1.3.6.1.4.1.5951.4.1.3.1.1.7)

Number of current client connections.

vsvrCurSrvrConnections (1.3.6.1.4.1.5951.4.1.3.1.1.8)

Number of current connections to the actual servers behind the virtual server.

vsvrSurgeCount (1.3.6.1.4.1.5951.4.1.3.1.1.10)

Number of requests in the surge queue.

vsvrTotalRequests (1.3.6.1.4.1.5951.4.1.3.1.1.30)

Total number of requests received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

vsvrTotalRequestBytes (1.3.6.1.4.1.5951.4.1.3.1.1.31)

Total number of request bytes received on this service or virtual server.

vsvrTotalResponses (1.3.6.1.4.1.5951.4.1.3.1.1.32)

Number of responses received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

vsvrTotalResponseBytes (1.3.6.1.4.1.5951.4.1.3.1.1.33)

Number of response bytes received by this service or virtual server.

vsvrTotalPktsRecvd (1.3.6.1.4.1.5951.4.1.3.1.1.34)

Total number of packets received by this service or virtual server.

vsvrTotalPktsSent (1.3.6.1.4.1.5951.4.1.3.1.1.35)

Total number of packets sent.

vsvrTotalSynsRecvd (1.3.6.1.4.1.5951.4.1.3.1.1.36)

Total number of SYN packets received from clients on this service (only when directly accessed) or virtual server.

vsvrCurServicesDown (1.3.6.1.4.1.5951.4.1.3.1.1.37)

The current number of services which are bound to this vserver and are in the state 'down'.

vsvrCurServicesUnKnown (1.3.6.1.4.1.5951.4.1.3.1.1.38)

The current number of services which are bound to this vserver and are in the state 'unKnown'.

vsvrCurServicesOutOfSvc (1.3.6.1.4.1.5951.4.1.3.1.1.39)

The current number of services which are bound to this vserver and are in the state 'outOfService'.

vsvrCurServicesTransToOutOfSvc (1.3.6.1.4.1.5951.4.1.3.1.1.40)

The current number of services which are bound to this vserver and are in the state 'transitionToOutOfService'.

vsvrCurServicesUp (1.3.6.1.4.1.5951.4.1.3.1.1.41)

The current number of services which are bound to this vserver and are in the state 'up'.

vsvrTotMiss (1.3.6.1.4.1.5951.4.1.3.1.1.42)

Total vserver misses

vsvrRequestRate (1.3.6.1.4.1.5951.4.1.3.1.1.43)

Request rate in requests per second for this service or virtual server.

vsvrRxBytesRate (1.3.6.1.4.1.5951.4.1.3.1.1.44)

Request rate in bytes per second for this service or virtual server.

vsvrTxBytesRate (1.3.6.1.4.1.5951.4.1.3.1.1.45)

Response rate in bytes per second for this service or virtual server.

vsvrSynfloodRate (1.3.6.1.4.1.5951.4.1.3.1.1.46)

Rate of unacknowledged SYN packets for this service or virtual server.

vsvrIp6Address (1.3.6.1.4.1.5951.4.1.3.1.1.47)

IPv6 address of the v server

vsvrTotHits (1.3.6.1.4.1.5951.4.1.3.1.1.48)

Total vserver hits

vsvrTotSpillOvers (1.3.6.1.4.1.5951.4.1.3.1.1.54)

Number of times vserver experienced spill over.

vsvrTotalClients (1.3.6.1.4.1.5951.4.1.3.1.1.56)

Total number of established client connections.

vsvrClientConnOpenRate (1.3.6.1.4.1.5951.4.1.3.1.1.58)

Rate at which connections are opened for this virtual server per second.

vsvrFullName (1.3.6.1.4.1.5951.4.1.3.1.1.59)

The name of the vserver

vsvrCurSslVpnUsers (1.3.6.1.4.1.5951.4.1.3.1.1.60)

Number of aaa sessions on this vserver

vsvrTotalServicesBound (1.3.6.1.4.1.5951.4.1.3.1.1.61)

The current number of services which are bound to this vserver.

vsvrHealth (1.3.6.1.4.1.5951.4.1.3.1.1.62)

The percentage of UP services bound to this vserver.

vsvrTicksSinceLastStateChange (1.3.6.1.4.1.5951.4.1.3.1.1.63)

Time (in 10 milliseconds) since the last state change.

vsvrEntityType (1.3.6.1.4.1.5951.4.1.3.1.1.64)

The type of the vserver.

vsvrTotalServers (1.3.6.1.4.1.5951.4.1.3.1.1.65)

Total number of established server connections.

vsvrActiveActiveState (1.3.6.1.4.1.5951.4.1.3.1.1.66)

The state of the vserver based on ActiveActive configuration.

vsvrInvalidRequestResponse (1.3.6.1.4.1.5951.4.1.3.1.1.67)

Number invalid requests/responses on this vserver

vsvrInvalidRequestResponseDropped (1.3.6.1.4.1.5951.4.1.3.1.1.68)

Number invalid requests/responses dropped on this vserver

vsvrTdid (1.3.6.1.4.1.5951.4.1.3.1.1.69)

Traffic Domain of the vserver

vsvrSoThreshold (1.3.6.1.4.1.5951.4.1.3.1.1.70)

Spill Over Threshold set on the VServer.

vsvrEstablishedConn (1.3.6.1.4.1.5951.4.1.3.1.1.71)

Number of client connections in ESTABLISHED state.

vsvrCurTotalVpnUsers (1.3.6.1.4.1.5951.4.1.3.1.1.72)

Number of total users on this vserver

vserverServiceTable (1.3.6.1.4.1.5951.4.1.3.2)

The vserver service relationship table

Indexed on: [vsvrName](#), [vsvrServiceName](#)

vsvrServiceHits (1.3.6.1.4.1.5951.4.1.3.2.1.5)

Number of times that the service has been provided.

servicePersistentHits (1.3.6.1.4.1.5951.4.1.3.2.1.6)

Total number of persistent hits.

serviceWeight (1.3.6.1.4.1.5951.4.1.3.2.1.7)

The weight of the service tied to the vserver.

vsvrServiceName (1.3.6.1.4.1.5951.4.1.3.2.1.8)

The name of the service to which the vserver is bound.

vsvrServiceFullName (1.3.6.1.4.1.5951.4.1.3.2.1.9)

The Full name of the service to which the vserver is bound.

vserverFullName (1.3.6.1.4.1.5951.4.1.3.2.1.10)

The full name of the vserver.

vsvrServiceEntityType (1.3.6.1.4.1.5951.4.1.3.2.1.11)

The entity type of the service: service group member or service.

vserverCspolicyTable (1.3.6.1.4.1.5951.4.1.3.3)

The vserver content switching policy relationship table for PE CS Policy

Indexed on: [vsvrName](#), [cspolicyName](#)

cspolicyName (1.3.6.1.4.1.5951.4.1.3.3.1.1)

This represents the name of the CS PE policy bound to content switching vserver

cspolicyDestVserverName (1.3.6.1.4.1.5951.4.1.3.3.1.2)

This represents the name of the destination vserver to which the request has to be directed to if the content switching policy evaluates to true.

cspolicyHits (1.3.6.1.4.1.5951.4.1.3.3.1.5)

The number of hits on this content switching policy.

csIndexVserverFullName (1.3.6.1.4.1.5951.4.1.3.3.1.6)

The full name of the cs vserver to which this policy belongs.

vserverCrpolicyTable (1.3.6.1.4.1.5951.4.1.3.4)

The vserver cache redirection policy relationship table

Indexed on: [vsrvName](#), [crpolicyName](#)

crpolicyName (1.3.6.1.4.1.5951.4.1.3.4.1.1)

This represents the name of the policy bound to cache-redirection vserver

crpolicyHits (1.3.6.1.4.1.5951.4.1.3.4.1.4)

Hits on the cache redirection policy.

crIndexVserverFullName (1.3.6.1.4.1.5951.4.1.3.4.1.5)

The full name of the cr vserver to which this policy belongs.

vserverGlobalStatsGroup (1.3.6.1.4.1.5951.4.1.3.5)

curConfigVservers (1.3.6.1.4.1.5951.4.1.3.5.1)

Total number of vservers configured on the NetScaler.

vsrvBindCount (1.3.6.1.4.1.5951.4.1.3.5.2)

Number of virtual server bindings on this NetScaler appliance.

vsrvSvcGrpBindCount (1.3.6.1.4.1.5951.4.1.3.5.3)

Number of virtual server, service group bindings on this NetScaler appliance.

curConfigLbVservers (1.3.6.1.4.1.5951.4.1.3.5.4)

Total number of LB vservers configured on the NetScaler.

curConfigGslbVservers (1.3.6.1.4.1.5951.4.1.3.5.5)

Total number of GSLB vservers configured on the NetScaler.

totSpilloverCount (1.3.6.1.4.1.5951.4.1.3.5.6)

Total count of spillovers.

lbvserverTable (1.3.6.1.4.1.5951.4.1.3.6)

Table for LB specific configuration

Indexed on: [vsrvName](#)

lbvsrvLBMethod (1.3.6.1.4.1.5951.4.1.3.6.1.1)

The Policy used for Load Balancing.

lbvsrvPersistenceType (1.3.6.1.4.1.5951.4.1.3.6.1.2)

The type of persistence used.

lbvsrvPersistenceTimeout (1.3.6.1.4.1.5951.4.1.3.6.1.3)

The timeout set for persistence.

lbvsvrActiveConn (1.3.6.1.4.1.5951.4.1.3.6.1.4)

Number of connections that are currently active.

lbvsvrAvgSvrTTFB (1.3.6.1.4.1.5951.4.1.3.6.1.5)

Average TTFB between the NetScaler appliance and the server. TTFB is the time interval between sending the request packet to a service and receiving the first response from the service

lbvsvrRdpCookieParsed (1.3.6.1.4.1.5951.4.1.3.6.1.6)

Number of times MSTS RDP Cookie got parsed on this vserver

vserverPqpolicyTable (1.3.6.1.4.1.5951.4.1.3.7)

The vserver priority queuing policy relationship table

Indexed on: vsvrName, pqName

pqpolicytotClientTransactionTime (1.3.6.1.4.1.5951.4.1.3.7.1.5)

Total client transaction time in micro-seconds for this priority queuing policy.

pqpolicytotClientTransactions (1.3.6.1.4.1.5951.4.1.3.7.1.6)

Total number of client transactions for this priority queuing policy.

pqpolycypqDropped (1.3.6.1.4.1.5951.4.1.3.7.1.7)

Total number of dropped transactions for this priority queuing policy.

pqpolycypqQdepth (1.3.6.1.4.1.5951.4.1.3.7.1.8)

Number of clients waiting currently for this priority queuing policy.

pqpolycypqAvgClientTransactionTime (1.3.6.1.4.1.5951.4.1.3.7.1.9)

Average time taken by a priority queuing client to complete its transaction for this priority queuing policy.

pqpolycypqVserverIP (1.3.6.1.4.1.5951.4.1.3.7.1.10)

IP address of the virtual server to which this priority queuing policy is bound.

pqpolycypqVserverPort (1.3.6.1.4.1.5951.4.1.3.7.1.11)

Port number of the virtual server to which this priority queuing policy is bound.

pqpolycypqCurrentClientConnections (1.3.6.1.4.1.5951.4.1.3.7.1.12)

Current number of server connections established for serving clients for this priority queuing policy.

pqpolycypqTotQueueDepth (1.3.6.1.4.1.5951.4.1.3.7.1.13)

Total number of waiting clients for this priority queuing policy.

pqpolycypqTotClientConnections (1.3.6.1.4.1.5951.4.1.3.7.1.14)

Total number of server connections established for serving clients for this priority queuing policy.

pqpolycypqTotQueueWaitTime (1.3.6.1.4.1.5951.4.1.3.7.1.15)

Amount of time spent by priority queuing clients waiting in the priority queue.

pqpolycypqTotAvgQueueDepth (1.3.6.1.4.1.5951.4.1.3.7.1.16)

Average number of waiting clients for this priority queuing policy.

pqpolycypqTotAvgQueueWaitTime (1.3.6.1.4.1.5951.4.1.3.7.1.17)

Average wait time for clients for this priority queuing policy.

pppolicytotClientTransactionTimems (1.3.6.1.4.1.5951.4.1.3.7.1.18)

Total client transaction time in microsec for this priority queuing policy.

pppolicyppqAvgClientTransactionTimems (1.3.6.1.4.1.5951.4.1.3.7.1.19)

Average time taken by a priority queuing client to complete its transaction for this priority queuing policy.

vserverScpolicyTable (1.3.6.1.4.1.5951.4.1.3.8)

The vserver sure connect policy relationship table

Indexed on: [svcServiceName](#), [scPolicyName](#)

vsrscpolicyPrimaryIPAddress (1.3.6.1.4.1.5951.4.1.3.8.1.1)

The IP address of the service or virtual server to which the policy is bound.

vsrscpolicyPrimaryPort (1.3.6.1.4.1.5951.4.1.3.8.1.2)

The port of the service or virtual server to which the policy is bound.

vsrscpolicydesIpAddress (1.3.6.1.4.1.5951.4.1.3.8.1.8)

IP address of the destination service.

vsrscpolicydestPort (1.3.6.1.4.1.5951.4.1.3.8.1.9)

Port number of the destination service.

vsrscpolicyavgServerTransactionTime (1.3.6.1.4.1.5951.4.1.3.8.1.10)

Average server transaction time in seconds for this SureConnect Policy.

vsrscpolicytotClientTransaction (1.3.6.1.4.1.5951.4.1.3.8.1.11)

Total number of client transactions processed by this SureConnect policy.

vsrscpolicytotOpenConn (1.3.6.1.4.1.5951.4.1.3.8.1.12)

Current number of open connections to the servers matching this policy.

vsrscpolycyscPhysicalServiceIP (1.3.6.1.4.1.5951.4.1.3.8.1.13)

IP address of the service for which these statistics are maintained.

vsrscpolycyscPhysicalServicePort (1.3.6.1.4.1.5951.4.1.3.8.1.14)

Port of the service for which these statistics are maintained.

vsrscpolycyscCurrentWaitingTime (1.3.6.1.4.1.5951.4.1.3.8.1.15)

Value of the currently estimated waiting time in seconds for the configured URL.

vsrscpolycyscCurrentClientConnections (1.3.6.1.4.1.5951.4.1.3.8.1.16)

Number of clients currently allowed a server connection by this SureConnect policy.

vsrscpolycyscTotalClientConnections (1.3.6.1.4.1.5951.4.1.3.8.1.17)

Total number of clients that were allowed a server connection by this SureConnect policy.

vsrscpolycyscTotalServerConnections (1.3.6.1.4.1.5951.4.1.3.8.1.18)

Total number of server connections that were established through this SureConnect policy.

vsrscpolycyscTotalRequestsReceived (1.3.6.1.4.1.5951.4.1.3.8.1.19)

Total number of requests received by this SureConnect policy.

vsvrscpolycyscTotalRequestBytes (1.3.6.1.4.1.5951.4.1.3.8.1.20)

Total number of request bytes received by this SureConnect policy.

vsvrscpolycyscTotalResponsesReceived (1.3.6.1.4.1.5951.4.1.3.8.1.21)

Total number of server responses received by this SureConnect policy.

vsvrscpolycyscTotalResponseBytes (1.3.6.1.4.1.5951.4.1.3.8.1.22)

Total number of response bytes received by this SureConnect policy.

vsvrscpolycyscCurrentSurgeQClients (1.3.6.1.4.1.5951.4.1.3.8.1.23)

Number of clients currently matching the SureConnect policy, but are in the surge queue.

vsvrscpolycyscCurrentWaitingClients (1.3.6.1.4.1.5951.4.1.3.8.1.24)

Current number of SureConnect priority clients that are waiting for a server connection.

vsvrscpolycyscTotalServerTransactions (1.3.6.1.4.1.5951.4.1.3.8.1.25)

Number of 200 OK responses received from the web server by this SureConnect policy.

vsvrscpolycyscTotalServerTTFBTransactions (1.3.6.1.4.1.5951.4.1.3.8.1.26)

Number of Time-To-First-Byte transactions from the web server for this SureConnect policy.

vsvrscpolycyscTotalServerTTLB (1.3.6.1.4.1.5951.4.1.3.8.1.27)

Server Time-To-Last-Byte in seconds calculated for this SureConnect policy.

vsvrscpolycyscTotalClientTTLB (1.3.6.1.4.1.5951.4.1.3.8.1.28)

Client Time-To-Last-Byte in seconds calculated for this SureConnect policy.

vsvrscpolycyscTotalServerTTFB (1.3.6.1.4.1.5951.4.1.3.8.1.29)

Server Time-To-First-Byte in seconds calculated for this SureConnect policy.

vsvrscpolycyscAverageClientTTLB (1.3.6.1.4.1.5951.4.1.3.8.1.30)

Average value of the client Time-To-Last-Byte in seconds for this SureConnect policy.

vsvrscpolycyscAverageServerTTFB (1.3.6.1.4.1.5951.4.1.3.8.1.31)

Average value of the server Time-To-First-Byte in seconds for this SureConnect policy.

vserverAdvanceSslConfigTable (1.3.6.1.4.1.5951.4.1.3.9)

The vserver advance SSL configuration

Indexed on: [vsvrName](#)

vsvrSslDH (1.3.6.1.4.1.5951.4.1.3.9.1.1)

Whether DH is enabled/disabled.

vsvrSslDHCount (1.3.6.1.4.1.5951.4.1.3.9.1.2)

The DH refresh count to re-generate public/private key.

vsvrSslDHFilePath (1.3.6.1.4.1.5951.4.1.3.9.1.3)

The DH file path name.

vsvrSslRSA (1.3.6.1.4.1.5951.4.1.3.9.1.4)

The ephemeral RSA support for service.

vsvrSslRSACount (1.3.6.1.4.1.5951.4.1.3.9.1.5)

The eRSA refresh count to re-generate RSA temporary key.

vsvrSslv2Protocol (1.3.6.1.4.1.5951.4.1.3.9.1.6)

The support for SSLv2 protocol for service.

vsvrSslv3Protocol (1.3.6.1.4.1.5951.4.1.3.9.1.7)

The support for SSLv3 protocol for service.

vsvrSslTLsv1Protocol (1.3.6.1.4.1.5951.4.1.3.9.1.8)

The support for TLSv1 protocol for service.

vsvrSslRedirectSupport (1.3.6.1.4.1.5951.4.1.3.9.1.9)

The support for ssl redirect for service.

vsvrSslClearTextPort (1.3.6.1.4.1.5951.4.1.3.9.1.10)

The clear text port on the backend webserver.

vserverCipherBindingTable (1.3.6.1.4.1.5951.4.1.3.10)

The vserver cipher bindings table

Indexed on: [vsvrName](#), [vsvrSslCipherBindName](#)

vsvrSslCipherBindName (1.3.6.1.4.1.5951.4.1.3.10.1.1)

The cipher name bound to this service.

vsvrSslCipherBindDesc (1.3.6.1.4.1.5951.4.1.3.10.1.2)

The Cipher description.

vserverCsPiPolicyTable (1.3.6.1.4.1.5951.4.1.3.11)

The vserver content switching policy relationship table for PI CS Policy

Indexed on: [vsvrName](#), [csPipolicyName](#)

csPipolicyName (1.3.6.1.4.1.5951.4.1.3.11.1.1)

This represents the name of the CS PI policy bound to content switching vserver

csPipolicyDestVserverName (1.3.6.1.4.1.5951.4.1.3.11.1.2)

This represents the name of the destination vserver to which the request has to be directed to if the content switching policy evaluates to true.

piPolicyBindingHits (1.3.6.1.4.1.5951.4.1.3.11.1.3)

Number of hits on the policy on this binding

csPiIndexVserverFullName (1.3.6.1.4.1.5951.4.1.3.11.1.4)

The full name of the cs vserver to which this policy belongs.

cspolicyActionName (1.3.6.1.4.1.5951.4.1.3.11.1.5)

The name of the CS PI policy action.

snmpTrapVarBindOidsGroup (1.3.6.1.4.1.5951.4.1.10.2)

alarmHighThreshold (1.3.6.1.4.1.5951.4.1.10.2.1)

This is the high threshold value configured for this alarm. When this threshold is crossed an SNMP alarm is generated.

alarmNormalThreshold (1.3.6.1.4.1.5951.4.1.10.2.2)

This is the normal threshold configured for this alarm which triggers the return-to-normal alarm.

entityName (1.3.6.1.4.1.5951.4.1.10.2.3)

This represents the name of the entity whose state has changed.

nsUserName (1.3.6.1.4.1.5951.4.1.10.2.4)

This represents the name of the system user.

configurationCmd (1.3.6.1.4.1.5951.4.1.10.2.5)

This represents the configuration command that was issued.

authorizationStatus (1.3.6.1.4.1.5951.4.1.10.2.6)

This represents the authorization status for an attempted configuration change.

commandExecutionStatus (1.3.6.1.4.1.5951.4.1.10.2.7)

This represents the command execution status for the attempted configuration change.

unackSynCount (1.3.6.1.4.1.5951.4.1.10.2.8)

The number of un-acknowledged SYNs NetScaler has received in the past synFlood time-interval.

alarmLowThreshold (1.3.6.1.4.1.5951.4.1.10.2.9)

This is the low threshold value configured for this alarm. When this threshold is crossed an SNMP alarm is generated.

alarmProbeFailedErrorString (1.3.6.1.4.1.5951.4.1.10.2.10)

This string represents the error occurred on the last monitor probe failure.

alarmVipRhiIpAddr (1.3.6.1.4.1.5951.4.1.10.2.11)

This represents the VIP whose RHI state has changed.

alarmVipRhiState (1.3.6.1.4.1.5951.4.1.10.2.12)

This represents the changed RHI state of the VIP.

alarmRateLmtThresholdExceeded (1.3.6.1.4.1.5951.4.1.10.2.13)

This specifies the name of the rate limit identifier that exceeded the threshold.

ipAddressGathered (1.3.6.1.4.1.5951.4.1.10.2.14)

This specifies the list of ip addresses that may have been gathered during the expression evaluation.

stringComputed (1.3.6.1.4.1.5951.4.1.10.2.15)

This contains the string computed during the expression evaluation.

alarmEntityCurState (1.3.6.1.4.1.5951.4.1.10.2.16)

This represents the state of vserver, physicalservice or servicegroup.

sysHealthPowerSupplyStatus (1.3.6.1.4.1.5951.4.1.10.2.17)

This text represents the status of power supply unit

alarmCurrentValue (1.3.6.1.4.1.5951.4.1.10.2.18)

This is the current value of the entity when high or normal threshold trap is sent.

alarmVipRhilNetAddressType (1.3.6.1.4.1.5951.4.1.10.2.19)

The address type of alarmVipRhilnetAddress

alarmVipRhilnetAddress (1.3.6.1.4.1.5951.4.1.10.2.20)

This represents the VIP whose RHI state has changed.

nsClientIPAddr (1.3.6.1.4.1.5951.4.1.10.2.23)

This represents the IP Address of the machine trying to access / connected to Netscaler.

ipConflictAddr (1.3.6.1.4.1.5951.4.1.10.2.24)

The IP configured in netscaler conflicting in the network.

appfwLogMsg (1.3.6.1.4.1.5951.4.1.10.2.25)

This represents the log message of appfw check violation.

dnskeyName (1.3.6.1.4.1.5951.4.1.10.2.26)

The name of the DNS key that is due for expiry.

dnskeyTimeToExpire (1.3.6.1.4.1.5951.4.1.10.2.27)

The amount of time for the key to expire

dnskeyUnitsOfExpiry (1.3.6.1.4.1.5951.4.1.10.2.28)

the units of the time of expiry

entityNewName (1.3.6.1.4.1.5951.4.1.10.2.29)

This represents the entity newName after name was changed.

entityOldName (1.3.6.1.4.1.5951.4.1.10.2.30)

This represents the entity Name before name was changed.

platformRateLimitPacketDropCount (1.3.6.1.4.1.5951.4.1.10.2.31)

This counter has the number of packets dropped due to platform rate limiting since the last check.

haLicenseMatchState (1.3.6.1.4.1.5951.4.1.10.2.32)

the state of HA License check

sslCardStatusMsg (1.3.6.1.4.1.5951.4.1.10.2.33)

This represents the interpretation details of sslCardStatus.

callHomeUploadEventStatusMsg (1.3.6.1.4.1.5951.4.1.10.2.34)

This represents the status of CallHome Upload Event.

oldCCOIP (1.3.6.1.4.1.5951.4.1.10.2.35)

This represents the old Configuration Coordinator IP.

newCCOIP (1.3.6.1.4.1.5951.4.1.10.2.36)

This represents the new Configuration Coordinator IP.

oldOVS (1.3.6.1.4.1.5951.4.1.10.2.37)

This represents the old cluster Operational View Set.

newOVS (1.3.6.1.4.1.5951.4.1.10.2.38)

This represents the new Cluster Operational View Set.

qosdVersion (1.3.6.1.4.1.5951.4.1.10.2.39)

This counter tells the QOSD version

brVersion (1.3.6.1.4.1.5951.4.1.10.2.40)

This counter tells the BR version

sslChipName (1.3.6.1.4.1.5951.4.1.10.2.41)

This represents the name of a Crypto device .

vrid (1.3.6.1.4.1.5951.4.1.10.2.42)

This represents the VRID.

vridBoundVIP (1.3.6.1.4.1.5951.4.1.10.2.43)

This represents the VRID bound VIP.

newVridPriority (1.3.6.1.4.1.5951.4.1.10.2.44)

This represents the new priority of VRID

effectiveVridPriority (1.3.6.1.4.1.5951.4.1.10.2.46)

This represents the effective priority of VRID

dstip (1.3.6.1.4.1.5951.4.1.10.2.47)

This is the dstip for which the port allocation has failed.

platformLicensedThroughput (1.3.6.1.4.1.5951.4.1.10.2.48)

This represents the platform licensed throughput.

platformLicensedPPS (1.3.6.1.4.1.5951.4.1.10.2.49)

This represents the platform licensed packets per seconds.

commandFailureReason (1.3.6.1.4.1.5951.4.1.10.2.50)

This is the error string displayed in cli when the command fails.

nsIPAddressType (1.3.6.1.4.1.5951.4.1.10.2.51)

The address type of nsIPAddress

nsIPAddress (1.3.6.1.4.1.5951.4.1.10.2.52)

This represents the IPV6 or IPV4 address.

Generic MIB-II Traps

coldStart (1.3.6.1.6.3.1.1.5.1)

A coldStart trap signifies that the SNMP entity, supporting a notification originator application, is reinitializing itself and that its configuration may have been altered.

linkDown (1.3.6.1.6.3.1.1.5.3)

A linkDown trap signifies that the SNMP entity, acting in an agent role, has detected that the ifOperStatus object for one of its communication links is about to enter the down

state from some other state (but not from the notPresent state). This other state is indicated by the included value of ifOperStatus.

Varbinds sent in the trap message: [ifIndex](#), [ifAdminStatus](#), [ifOperStatus](#)

linkUp (1.3.6.1.6.3.1.1.5.4)

A linkUp trap signifies that the SNMP entity, acting in an agent role, has detected that the ifOperStatus object for one of its communication links left the down state and transitioned into some other state (but not into the notPresent state). This other state is indicated by the included value of ifOperStatus.

Varbinds sent in the trap message: [ifIndex](#), [ifAdminStatus](#), [ifOperStatus](#)

authenticationFailure (1.3.6.1.6.3.1.1.5.5)

An authenticationFailure trap signifies that the SNMP entity has received a protocol message that is not properly authenticated. While all implementations of SNMP entities MAY be capable of generating this trap, the snmpEnableAuthenTraps object indicates whether this trap will be generated.

NetScaler Enterprise Traps

changeToPrimary (1.3.6.1.4.1.5951.1.1.0.1)

This trap indicates that the netscaler is now operating in the primary mode.

Varbinds sent in the trap message: [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: HA-STATE-CHANGE

changeToSecondary (1.3.6.1.4.1.5951.1.1.0.2)

This trap indicates that the netscaler is now operating in the Secondary mode.

Varbinds sent in the trap message: [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: HA-STATE-CHANGE

cpuUtilization (1.3.6.1.4.1.5951.1.1.0.3)

This trap indicates that the CPU utilization has exceeded the high threshold

Varbinds sent in the trap message: [nsCPUUsage](#), [alarmHighThreshold](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: CPU-USAGE

entitydown (1.3.6.1.4.1.5951.1.1.0.8)

This trap is sent when the state of entities such as an interface, vserver, physicalservice or servicegroup changes to DOWN

Varbinds sent in the trap message: [entityName](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: ENTITY-STATE

entityup (1.3.6.1.4.1.5951.1.1.0.9)

This trap is sent when the state of entities such as an interface, vserver, physicalservice or servicegroup changes to UP

Varbinds sent in the trap message: [entityName](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: ENTITY-STATE

synflood (1.3.6.1.4.1.5951.1.1.0.10)

This trap is sent when the rate at which unacknowledged SYNs are received cross a threshold value

Varbinds sent in the trap message: [unackSynCount](#), [alarmHighThreshold](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: SYNFLOOD

cpuUtilizationNormal (1.3.6.1.4.1.5951.1.1.0.11)

This trap indicates that the CPU utilization has come back to normal

Varbinds sent in the trap message: [nsCPUUsage](#), [alarmNormalThreshold](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: CPU-USAGE

synfloodNormal (1.3.6.1.4.1.5951.1.1.0.12)

This trap is sent when the rate at which unacknowledged SYNs are received returns to normal

Varbinds sent in the trap message: [unackSynCount](#), [alarmNormalThreshold](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: SYNFLOOD

memoryUtilization (1.3.6.1.4.1.5951.1.1.0.13)

This trap is sent when the memory utilization of the system exceeds the threshold value

Varbinds sent in the trap message: [resMemUsage](#), [alarmHighThreshold](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: MEMORY

memoryUtilizationNormal (1.3.6.1.4.1.5951.1.1.0.14)

This trap is sent when the memory utilization of the system returns to normal

Varbinds sent in the trap message: [resMemUsage](#), [alarmNormalThreshold](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: MEMORY

vServerRequestRate (1.3.6.1.4.1.5951.1.1.0.15)

This trap is sent when the request rate on a vServer exceeds a threshold value

Varbinds sent in the trap message: [vsvrName](#), [vsvrRequestRate](#), [alarmHighThreshold](#), [vsvrFullName](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: VSERVER-REQRATE

vServerRequestRateNormal (1.3.6.1.4.1.5951.1.1.0.16)

This trap is sent when the request rate on a vServer returns to normal

Varbinds sent in the trap message: [vsvrName](#), [vsvrRequestRate](#), [alarmNormalThreshold](#), [vsvrFullName](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: VSERVER-REQRATE

serviceRequestRate (1.3.6.1.4.1.5951.1.1.0.17)

This trap is sent when the request rate on a service exceeds a threshold value

Varbinds sent in the trap message: `svcServiceName`, `svcRequestRate`, `alarmHighThreshold`, `svcServiceFullName`, `sysIpAddress`

To receive this trap, enable the following SNMP alarm: SERVICE-REQRATE

serviceRequestRateNormal (1.3.6.1.4.1.5951.1.1.0.18)

This trap is sent when the request rate on a service returns to normal

Varbinds sent in the trap message: `svcServiceName`, `svcRequestRate`, `alarmNormalThreshold`, `svcServiceFullName`, `sysIpAddress`

To receive this trap, enable the following SNMP alarm: SERVICE-REQRATE

netScalerConfigChange (1.3.6.1.4.1.5951.1.1.0.25)

This trap is sent when the configuration on the NetScaler is changed.

Varbinds sent in the trap message: `nsUserName`, `configurationCmd`, `authorizationStatus`, `commandExecutionStatus`, `nsClientIPAddr`, `sysIpAddress`

To receive this trap, enable the following SNMP alarm: CONFIG-CHANGE

maxClients (1.3.6.1.4.1.5951.1.1.0.26)

This trap is sent when the number of clients hits the `maxClients` value for a service

Varbinds sent in the trap message: `svcServiceName`, `svcEstablishedConn`, `alarmHighThreshold`, `svcServiceFullName`, `sysIpAddress`

To receive this trap, enable the following SNMP alarm: SERVICE-MAXCLIENTS

maxClientsNormal (1.3.6.1.4.1.5951.1.1.0.27)

This trap is sent when the number of clients falls below 70% of `maxClients` value for a service.

Varbinds sent in the trap message: `svcServiceName`, `svcEstablishedConn`, `alarmNormalThreshold`, `svcServiceFullName`, `sysIpAddress`

To receive this trap, enable the following SNMP alarm: SERVICE-MAXCLIENTS

netScalerConfigSave (1.3.6.1.4.1.5951.1.1.0.28)

This trap is sent when the configuration on the NetScaler is saved.

Varbinds sent in the trap message: `nsUserName`, `sysIpAddress`

To receive this trap, enable the following SNMP alarm: CONFIG-SAVE

serviceRxBytesRate (1.3.6.1.4.1.5951.1.1.0.29)

This trap is sent when the request bytes/s of a service exceeds a threshold value.

Varbinds sent in the trap message: `svcServiceName`, `svcRxBytesRate`, `alarmHighThreshold`, `svcServiceFullName`, `sysIpAddress`

To receive this trap, enable the following SNMP alarm: ENTITY-RXRATE

serviceRxBytesRateNormal (1.3.6.1.4.1.5951.1.1.0.30)

This trap is sent when the request bytes/s of a service returns to normal.

Varbinds sent in the trap message: `svcServiceName`, `svcRxBytesRate`, `alarmNormalThreshold`, `svcServiceFullName`, `sysIpAddress`

To receive this trap, enable the following SNMP alarm: ENTITY-RXRATE

vserverRxBytesRate (1.3.6.1.4.1.5951.1.1.0.31)

This trap is sent when the request bytes/s of a vserver exceeds a threshold value.

Varbinds sent in the trap message: [vsvrName](#), [vsvrRxBytesRate](#), [alarmHighThreshold](#), [vsvrFullName](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: ENTITY-RXRATE

vserverRxBytesRateNormal (1.3.6.1.4.1.5951.1.1.0.32)

This trap is sent when the request bytes/s of a vServer returns to normal.

Varbinds sent in the trap message: [vsvrName](#), [vsvrRxBytesRate](#), [alarmNormalThreshold](#), [vsvrFullName](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: ENTITY-RXRATE

serviceTxBytesRate (1.3.6.1.4.1.5951.1.1.0.33)

This trap is sent when the response bytes/s of a service exceeds a threshold value.

Varbinds sent in the trap message: [svcServiceName](#), [svcTxBytesRate](#), [alarmHighThreshold](#), [svcServiceFullName](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: ENTITY-TXRATE

serviceTxBytesRateNormal (1.3.6.1.4.1.5951.1.1.0.34)

This trap is sent when the response bytes/s of a service returns to normal.

Varbinds sent in the trap message: [svcServiceName](#), [svcTxBytesRate](#), [alarmNormalThreshold](#), [svcServiceFullName](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: ENTITY-TXRATE

vserverTxBytesRate (1.3.6.1.4.1.5951.1.1.0.35)

This trap is sent when the response bytes/s of a vserver exceeds a threshold value.

Varbinds sent in the trap message: [vsvrName](#), [vsvrTxBytesRate](#), [alarmHighThreshold](#), [vsvrFullName](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: ENTITY-TXRATE

vserverTxBytesRateNormal (1.3.6.1.4.1.5951.1.1.0.36)

This trap is sent when the response bytes/s of a vServer returns to normal.

Varbinds sent in the trap message: [vsvrName](#), [vsvrTxBytesRate](#), [alarmNormalThreshold](#), [vsvrFullName](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: ENTITY-TXRATE

serviceSynfloodRate (1.3.6.1.4.1.5951.1.1.0.37)

This trap is sent when the number of unacknowledged syns for a service exceeds a threshold value.

Varbinds sent in the trap message: [svcServiceName](#), [svcSynfloodRate](#), [alarmHighThreshold](#), [svcServiceFullName](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: ENTITY-SYNFLOOD

serviceSynfloodNormal (1.3.6.1.4.1.5951.1.1.0.38)

This trap is sent when the number of unacknowledged syns for a service returns to normal.

Varbinds sent in the trap message: [svcServiceName](#), [svcSynfloodRate](#), [alarmNormalThreshold](#), [svcServiceFullName](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: ENTITY-SYNFLOOD

vserverSynfloodRate (1.3.6.1.4.1.5951.1.1.0.39)

This trap is sent when the number of unacknowledged syns for a vserver exceeds a threshold value.

Varbinds sent in the trap message: [vsvrName](#), [vsvrSynfloodRate](#), [alarmHighThreshold](#), [vsvrFullName](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: ENTITY-SYNFLOOD

vserverSynfloodNormal (1.3.6.1.4.1.5951.1.1.0.40)

This trap is sent when the number of unacknowledged syns for a vserver returns to normal.

Varbinds sent in the trap message: `vsvrName`, `vsvrSynfloodRate`, `alarmNormalThreshold`, `vsvrFullName`, `sysIpAddress`

To receive this trap, enable the following SNMP alarm: ENTITY-SYNFLOOD

svcGroupMemberRequestRate (1.3.6.1.4.1.5951.1.1.0.41)

This trap is sent when the request rate on a service group member exceeds a threshold value

Varbinds sent in the trap message: `svcGrpMemberName`, `svcGrpMemberRequestRate`, `alarmHighThreshold`, `svcGrpMemberFullName`, `sysIpAddress`

To receive this trap, enable the following SNMP alarm: SERVICEGROUP-MEMBER-REQRATE

svcGroupMemberRequestRateNormal (1.3.6.1.4.1.5951.1.1.0.42)

This trap is sent when the request rate on a service group member returns to normal

Varbinds sent in the trap message: `svcGrpMemberName`, `svcGrpMemberRequestRate`, `alarmNormalThreshold`, `svcGrpMemberFullName`, `sysIpAddress`

To receive this trap, enable the following SNMP alarm: SERVICEGROUP-MEMBER-REQRATE

svcGroupMemberRxBytesRate (1.3.6.1.4.1.5951.1.1.0.43)

This trap is sent when the request bytes/s of a service group exceeds a threshold value.

Varbinds sent in the trap message: `svcGrpMemberName`, `svcGrpMemberRxBytesRate`, `alarmHighThreshold`, `svcGrpMemberFullName`, `sysIpAddress`

To receive this trap, enable the following SNMP alarm: ENTITY-RXRATE

svcGroupMemberRxBytesRateNormal (1.3.6.1.4.1.5951.1.1.0.44)

This trap is sent when the request bytes/s of a service group returns to normal.

Varbinds sent in the trap message: `svcGrpMemberName`, `svcGrpMemberRxBytesRate`, `alarmNormalThreshold`, `svcGrpMemberFullName`, `sysIpAddress`

To receive this trap, enable the following SNMP alarm: ENTITY-RXRATE

svcGroupMemberTxBytesRate (1.3.6.1.4.1.5951.1.1.0.45)

This trap is sent when the response bytes/s of a service group exceeds a threshold value.

Varbinds sent in the trap message: `svcGrpMemberName`, `svcGrpMemberTxBytesRate`, `alarmHighThreshold`, `svcGrpMemberFullName`, `sysIpAddress`

To receive this trap, enable the following SNMP alarm: ENTITY-TXRATE

svcGroupMemberTxBytesRateNormal (1.3.6.1.4.1.5951.1.1.0.46)

This trap is sent when the response bytes/s of a service group returns to normal.

Varbinds sent in the trap message: `svcGrpMemberName`, `svcGrpMemberTxBytesRate`, `alarmNormalThreshold`, `svcGrpMemberFullName`, `sysIpAddress`

To receive this trap, enable the following SNMP alarm: ENTITY-TXRATE

svcGroupMemberSynfloodRate (1.3.6.1.4.1.5951.1.1.0.47)

This trap is sent when the number of unacknowledged syns for a service group exceeds a threshold value.

Varbinds sent in the trap message: `svcGrpMemberName`, `svcGrpMemberSynfloodRate`, `alarmHighThreshold`, `svcGrpMemberFullName`, `sysIpAddress`

To receive this trap, enable the following SNMP alarm: ENTITY-SYNFLOOD

svcGroupMemberSynfloodNormal (1.3.6.1.4.1.5951.1.1.0.48)

This trap is sent when the number of unacknowledged syns for a service group returns to normal.

Varbinds sent in the trap message: [svcGrpMemberName](#), [svcGrpMemberSynfloodRate](#), [alarmNormalThreshold](#), [svcGrpMemberFullName](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: ENTITY-SYNFLOOD

svcGroupMemberMaxClients (1.3.6.1.4.1.5951.1.1.0.49)

This trap is sent when the number of clients hits the maxClients value for a service group member

Varbinds sent in the trap message: [svcGrpMemberName](#), [svcGrpMemberEstablishedConn](#), [alarmHighThreshold](#), [svcGrpMemberFullName](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: SERVICEGROUP-MEMBER-MAXCLIENTS

svcGroupMemberMaxClientsNormal (1.3.6.1.4.1.5951.1.1.0.50)

This trap is sent when the number of clients falls below 70% of maxClients value for a service group member.

Varbinds sent in the trap message: [svcGrpMemberName](#), [svcGrpMemberEstablishedConn](#), [alarmNormalThreshold](#), [svcGrpMemberFullName](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: SERVICEGROUP-MEMBER-MAXCLIENTS

averageCpuUtilization (1.3.6.1.4.1.5951.1.1.0.51)

This trap indicates that the average CPU usage in the multi-processor NetScaler system has exceeded the high threshold.

Varbinds sent in the trap message: [resCpuUsage](#), [alarmHighThreshold](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: AVERAGE-CPU

averageCpuUtilizationNormal (1.3.6.1.4.1.5951.1.1.0.52)

This trap indicates that the average CPU usage in the multi-processor NetScaler system has come back to normal.

Varbinds sent in the trap message: [resCpuUsage](#), [alarmNormalThreshold](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: AVERAGE-CPU

monRespTimeoutAboveThresh (1.3.6.1.4.1.5951.1.1.0.53)

This trap is sent when the response timeout for a monitor probe exceeds the configured threshold.

Varbinds sent in the trap message: [monServiceName](#), [monitorName](#), [responseTimeoutThreshold](#), [alarmMonrespto](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: MONITOR-RTO-THRESHOLD

monRespTimeoutBelowThresh (1.3.6.1.4.1.5951.1.1.0.54)

This trap is sent when the response timeout for a monitor probe comes back to normal, less than the threshold set.

Varbinds sent in the trap message: [monServiceName](#), [monitorName](#), [responseTimeoutThreshold](#), [alarmMonrespto](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: MONITOR-RTO-THRESHOLD

netScalerLoginFailure (1.3.6.1.4.1.5951.1.1.0.55)

This trap is sent when a login attempt to the NetScaler fails.

Varbinds sent in the trap message: [nsUserName](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: LOGIN-FAILURE

sslCertificateExpiry (1.3.6.1.4.1.5951.1.1.0.56)

This trap is sent as an advance notification when an SSL certificate is due to expire.

Varbinds sent in the trap message: [sslCertKeyName](#), [sslDaysToExpire](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: SSL-CERT-EXPIRY

fanSpeedLow (1.3.6.1.4.1.5951.1.1.0.57)

This trap indicates that a fan speed has gone below an alarm threshold.

Varbinds sent in the trap message: [sysHealthCounterName](#), [sysHealthCounterValue](#), [alarmLowThreshold](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: FAN-SPEED-LOW

fanSpeedNormal (1.3.6.1.4.1.5951.1.1.0.58)

This trap indicates that a fan speed has returned to normal.

Varbinds sent in the trap message: [sysHealthCounterName](#), [sysHealthCounterValue](#), [alarmNormalThreshold](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: FAN-SPEED-LOW

voltageLow (1.3.6.1.4.1.5951.1.1.0.59)

This trap indicates that a voltage has gone low.

Varbinds sent in the trap message: [sysHealthCounterName](#), [sysHealthCounterValue](#), [alarmLowThreshold](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: VOLTAGE-LOW

voltageNormal (1.3.6.1.4.1.5951.1.1.0.60)

This trap indicates that a voltage has returned to normal.

Varbinds sent in the trap message: [sysHealthCounterName](#), [sysHealthCounterValue](#), [alarmNormalThreshold](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: VOLTAGE-LOW

voltageHigh (1.3.6.1.4.1.5951.1.1.0.61)

This trap indicates that a voltage has gone high.

Varbinds sent in the trap message: [sysHealthCounterName](#), [sysHealthCounterValue](#), [alarmHighThreshold](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: VOLTAGE-HIGH

temperatureHigh (1.3.6.1.4.1.5951.1.1.0.62)

This trap indicates that a temperature has gone high.

Varbinds sent in the trap message: [sysHealthCounterName](#), [sysHealthCounterValue](#), [alarmHighThreshold](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: TEMPERATURE-HIGH

temperatureNormal (1.3.6.1.4.1.5951.1.1.0.63)

This trap indicates that a temperature has returned to normal.

Varbinds sent in the trap message: [sysHealthCounterName](#), [sysHealthCounterValue](#), [alarmNormalThreshold](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: TEMPERATURE-HIGH

diskUsageHigh (1.3.6.1.4.1.5951.1.1.0.64)

This trap indicates that disk usage has gone high.

Varbinds sent in the trap message: [sysHealthDiskName](#), [sysHealthDiskPerusage](#), [alarmHighThreshold](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: DISK-USAGE-HIGH

diskUsageNormal (1.3.6.1.4.1.5951.1.1.0.65)

This trap indicates that disk usage has returned to normal.

Varbinds sent in the trap message: [sysHealthDiskName](#), [sysHealthDiskPerusage](#), [alarmNormalThreshold](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: DISK-USAGE-HIGH

interfaceThroughputLow (1.3.6.1.4.1.5951.1.1.0.66)

This trap indicates that interface throughput is low.

Varbinds sent in the trap message: [ifName](#), [ifThroughput](#), [ifMinThroughput](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: INTERFACE-THROUGHPUT-LOW

interfaceThroughputNormal (1.3.6.1.4.1.5951.1.1.0.67)

This trap indicates that interface throughput has returned to normal.

Varbinds sent in the trap message: [ifName](#), [ifThroughput](#), [ifMinThroughput](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: INTERFACE-THROUGHPUT-LOW

haVersionMismatch (1.3.6.1.4.1.5951.1.1.0.68)

This trap indicates that there is a mismatch in the OS version of the netscalers participating in HA.

Varbinds sent in the trap message: [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: HA-VERSION-MISMATCH

haSyncFailure (1.3.6.1.4.1.5951.1.1.0.69)

This trap indicates that config synchronization has failed on secondary.

Varbinds sent in the trap message: [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: HA-SYNC-FAILURE

haNoHeartbeats (1.3.6.1.4.1.5951.1.1.0.70)

This trap indicates that HA heartbeats are not received from the secondary.

Varbinds sent in the trap message: [haNicsMonitorFailed](#), [haLastNicMonitorFailed](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: HA-NO-HEARTBEATS

haBadSecState (1.3.6.1.4.1.5951.1.1.0.71)

This trap indicates that the secondary is in DOWN/UNKNOWN/STAY SECONDARY state.

Varbinds sent in the trap message: [haPeerSystemState](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: HA-BAD-SECONDARY-STATE

interfaceBWUseHigh (1.3.6.1.4.1.5951.1.1.0.72)

This trap is sent when the bandwidth usage of any of the interfaces of the system exceeds the threshold value (configured in Mbits/second)

Varbinds sent in the trap message: [ifName](#), [alarmHighThreshold](#), [alarmCurrentValue](#), [platformLicensedThroughput](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: INTERFACE-BW-USAGE

interfaceBWUseNormal (1.3.6.1.4.1.5951.1.1.0.73)

This trap is sent when the bandwidth usage of any of the interfaces of the system returns to normal

Varbinds sent in the trap message: [ifName](#), [alarmNormalThreshold](#), [alarmCurrentValue](#), [platformLicensedThroughput](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: INTERFACE-BW-USAGE

aggregateBWUseHigh (1.3.6.1.4.1.5951.1.1.0.74)

This trap is sent when the aggregate bandwidth usage of the system exceeds the threshold value (configured in Mbits/second)

Varbinds sent in the trap message: [alarmHighThreshold](#), [alarmCurrentValue](#), [platformLicensedThroughput](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: INTERFACE-BW-USAGE

aggregateBWUseNormal (1.3.6.1.4.1.5951.1.1.0.75)

This trap is sent when the aggregate bandwidth usage of the system returns to normal.

Varbinds sent in the trap message: [alarmNormalThreshold](#), [alarmCurrentValue](#), [platformLicensedThroughput](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: INTERFACE-BW-USAGE

vserverRhiStateChange (1.3.6.1.4.1.5951.1.1.0.76)

This trap is sent when the vserver RHI state changes.

Varbinds sent in the trap message: [alarmVipRhiState](#), [alarmVipRhilnetAddressType](#), [alarmVipRhilnetAddress](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: ENTITY-STATE

rateLmtThresholdExceed (1.3.6.1.4.1.5951.1.1.0.77)

This trap is sent when the client exceeds the ratelimit threshold.

Varbinds sent in the trap message: [alarmRateLmtThresholdExceeded](#), [ipAddressGathered](#), [stringComputed](#), [platformLicensedThroughput](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: RATE-LIMIT-THRESHOLD-EXCEEDED

monProbeFailed (1.3.6.1.4.1.5951.1.1.0.78)

This trap is sent when the monitor probe fails for configured number of retries in given max retries attempts.

Varbinds sent in the trap message: [monServiceName](#), [monitorName](#), [alarmProbeFailedRetries](#), [monitorRetrys](#), [alarmProbeFailedErrorString](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: MON_PROBE_FAILED

temperatureCpuHigh (1.3.6.1.4.1.5951.1.1.0.79)

This trap indicates that a CPU temperature has gone high.

Varbinds sent in the trap message: [sysHealthCounterName](#), [sysHealthCounterValue](#), [alarmHighThreshold](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: CPU-TEMPERATURE-HIGH

temperatureCpuNormal (1.3.6.1.4.1.5951.1.1.0.80)

This trap indicates that a CPU temperature has returned to normal.

Varbinds sent in the trap message: [sysHealthCounterName](#), [sysHealthCounterValue](#), [alarmNormalThreshold](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: CPU-TEMPERATURE-HIGH

entityofs (1.3.6.1.4.1.5951.1.1.0.81)

This trap is sent when the state of entities such as vserver, physicalservice or servicegroup changes to OUT OF SERVICE

Varbinds sent in the trap message: [entityName](#), [alarmEntityCurState](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: ENTITY-STATE

powerSupplyFailed (1.3.6.1.4.1.5951.1.1.0.82)

This trap is sent when power supply has failed or disconnected from the system

Varbinds sent in the trap message: [sysHealthCounterName](#), [sysHealthCounterValue](#), [sysHealthPowerSupplyStatus](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: POWER-SUPPLY-FAILURE

powerSupplyNormal (1.3.6.1.4.1.5951.1.1.0.83)

This trap is sent when power supply status returned back to normal

Varbinds sent in the trap message: [sysHealthCounterName](#), [sysHealthCounterValue](#), [sysHealthPowerSupplyStatus](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: POWER-SUPPLY-FAILURE

entityNameChanged (1.3.6.1.4.1.5951.1.1.0.84)

This trap is sent when vserver/service/sgroup/lbgroup/server entity is renamed

Varbinds sent in the trap message: [entityName](#), [entityOldName](#), [entityNewName](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: ENTITY-NAME-CHANGE

haPropFailure (1.3.6.1.4.1.5951.1.1.0.85)

This trap indicates that config propagation has failed on secondary.

Varbinds sent in the trap message: [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: HA-PROP-FAILURE

ipConflict (1.3.6.1.4.1.5951.1.1.0.86)

This trap indicates that ip conflict is present with another device in the network.

Varbinds sent in the trap message: [ipConflictAddr](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: IP-CONFLICT

appfwStartUrl (1.3.6.1.4.1.5951.1.1.0.87)

This trap indicates that AppFirewall Start URL violation occurred.

Varbinds sent in the trap message: [appfwLogMsg](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: APPFW-START-URL

appfwDenyUrl (1.3.6.1.4.1.5951.1.1.0.88)

This trap indicates that AppFirewall Deny URL violation occurred.

Varbinds sent in the trap message: [appfwLogMsg](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: APPFW-DENY-URL

appfwRefererHeader (1.3.6.1.4.1.5951.1.1.0.89)

This trap indicates that AppFirewall Referer Header violation occurred.

Varbinds sent in the trap message: [appfwLogMsg](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: APPFW-REFERER-HEADER

appfwCSRFtag (1.3.6.1.4.1.5951.1.1.0.90)

This trap indicates that AppFirewall CSRF Tag violation occurred.

Varbinds sent in the trap message: [appfwLogMsg](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: APPFW-CSRF-TAG

appfwCookie (1.3.6.1.4.1.5951.1.1.0.91)

This trap indicates that AppFirewall Cookie violation occurred.

Varbinds sent in the trap message: [appfwLogMsg](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: APPFW-COOKIE

appfwFieldConsistency (1.3.6.1.4.1.5951.1.1.0.92)

This trap indicates that AppFirewall Field Consistency violation occurred.

Varbinds sent in the trap message: [appfwLogMsg](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: APPFW-FIELD-CONSISTENCY

appfwBufferOverflow (1.3.6.1.4.1.5951.1.1.0.93)

This trap indicates that AppFirewall Buffer Overflow violation occurred.

Varbinds sent in the trap message: [appfwLogMsg](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: APPFW-BUFFER-OVERFLOW

appfwFieldFormat (1.3.6.1.4.1.5951.1.1.0.94)

This trap indicates that AppFirewall Field Format violation occurred.

Varbinds sent in the trap message: [appfwLogMsg](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: APPFW-FIELD-FORMAT

appfwSafeCommerce (1.3.6.1.4.1.5951.1.1.0.95)

This trap indicates that AppFirewall Safe Commerce violation occurred.

Varbinds sent in the trap message: [appfwLogMsg](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: APPFW-SAFE-COMMERCE

appfwSafeObject (1.3.6.1.4.1.5951.1.1.0.96)

This trap indicates that AppFirewall Safe Object violation occurred.

Varbinds sent in the trap message: [appfwLogMsg](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: APPFW-SAFE-OBJECT

appfwPolicyHit (1.3.6.1.4.1.5951.1.1.0.97)

This trap indicates that AppFirewall Policy Hit occurred.

Varbinds sent in the trap message: [appfwLogMsg](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: APPFW-POLICY-HIT

appfwXSS (1.3.6.1.4.1.5951.1.1.0.98)

This trap indicates that AppFirewall Cross Site Scripting violation occurred.

Varbinds sent in the trap message: [appfwLogMsg](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: APPFW-XSS

appfwXMLXSS (1.3.6.1.4.1.5951.1.1.0.99)

This trap indicates that AppFirewall XML Cross Site Scripting violation occurred.

Varbinds sent in the trap message: [appfwLogMsg](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: APPFW-XML-XSS

appfwSQL (1.3.6.1.4.1.5951.1.1.0.100)

This trap indicates that AppFirewall SQL violation occurred.

Varbinds sent in the trap message: [appfwLogMsg](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: APPFW-SQL

appfwXMLSQL (1.3.6.1.4.1.5951.1.1.0.101)

This trap indicates that AppFirewall XML SQL violation occurred.

Varbinds sent in the trap message: [appfwLogMsg](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: APPFW-XML-SQL

appfwXMLAttachment (1.3.6.1.4.1.5951.1.1.0.102)

This trap indicates that AppFirewall XML Attachment violation occurred.

Varbinds sent in the trap message: [appfwLogMsg](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: APPFW-XML-ATTACHMENT

appfwXMLDos (1.3.6.1.4.1.5951.1.1.0.103)

This trap indicates that AppFirewall XML DoS violation occurred.

Varbinds sent in the trap message: [appfwLogMsg](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: APPFW-XML-DOS

appfwXMLValidation (1.3.6.1.4.1.5951.1.1.0.104)

This trap indicates that AppFirewall XML Validation violation occurred.

Varbinds sent in the trap message: [appfwLogMsg](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: APPFW-XML-VALIDATION

appfwXMLWSI (1.3.6.1.4.1.5951.1.1.0.105)

This trap indicates that AppFirewall XML WSI violation occurred.

Varbinds sent in the trap message: [appfwLogMsg](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: APPFW-XML-WSI

appfwXMLSchemaCompile (1.3.6.1.4.1.5951.1.1.0.106)

This trap indicates that AppFirewall XML Schema Compile violation occurred.

Varbinds sent in the trap message: [appfwLogMsg](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: APPFW-XML-SCHEMA-COMPILE

appfwXMLSoapFault (1.3.6.1.4.1.5951.1.1.0.107)

This trap indicates that AppFirewall XML Soap Fault violation occurred.

Varbinds sent in the trap message: [appfwLogMsg](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: APPFW-XML-SOAP-FAULT

dnskeyExpiry (1.3.6.1.4.1.5951.1.1.0.108)

This trap is sent as an advance notification when an DNSKEY is due to expire.

Varbinds sent in the trap message: [dnskeyName](#), [dnskeyTimeToExpire](#), [dnskeyUnitsOfExpiry](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: DNSKEY-EXPIRY

platformRateLimitThresholdHigh (1.3.6.1.4.1.5951.1.1.0.109)

This trap indicates that the platform rate limit (in Mbps) has exceeded the threshold

Varbinds sent in the trap message: [alarmHighThreshold](#), [alarmCurrentValue](#), [platformLicensedThroughput](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: PF-RL-RATE-THRESHOLD

platformRateLimitThresholdNormal (1.3.6.1.4.1.5951.1.1.0.110)

This trap indicates that the platform rate limit (in Mbps) has come back to normal

Varbinds sent in the trap message: [alarmNormalThreshold](#), [alarmCurrentValue](#), [platformLicensedThroughput](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: PF-RL-RATE-THRESHOLD

platformPpsLimitThresholdHigh (1.3.6.1.4.1.5951.1.1.0.111)

This trap indicates that the platform packets per second (pps) limit has exceeded the threshold

Varbinds sent in the trap message: [alarmHighThreshold](#), [alarmCurrentValue](#), [platformLicensedPPS](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: PF-RL-PPS-THRESHOLD

platformPpsLimitThresholdNormal (1.3.6.1.4.1.5951.1.1.0.112)

This trap indicates that the platform packets per second (pps) limit has come back to normal

Varbinds sent in the trap message: [alarmNormalThreshold](#), [alarmCurrentValue](#), [platformLicensedPPS](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: PF-RL-PPS-THRESHOLD

platformRateLimitPktDrop (1.3.6.1.4.1.5951.1.1.0.113)

This trap is sent when packets are dropped due to platform rate limit (in Mbps) being reached

Varbinds sent in the trap message: [platformRateLimitPacketDropCount](#), [platformLicensedThroughput](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: PF-RL-RATE-PKTS-DROPPED

platformPpsLimitPktDrop (1.3.6.1.4.1.5951.1.1.0.114)

This trap is sent when packets are dropped due to platform packets per second (pps) limit being reached

Varbinds sent in the trap message: [platformRateLimitPacketDropCount](#), [platformLicensedPPS](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: PF-RL-PPS-PKTS-DROPPED

DataStreamRateLimitHit (1.3.6.1.4.1.5951.1.1.0.115)

DataStream Rate-Limiting is Removed. So, this trap is not required.

haLicenseCheck (1.3.6.1.4.1.5951.1.1.0.116)

This trap is sent when the NetScaler comes up and tells the state HA license check whether it is matched or mismatched

Varbinds sent in the trap message: [haLicenseMatchState](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: HA-LICENSE-MISMATCH

sslCardFailed (1.3.6.1.4.1.5951.1.1.0.117)

This trap is sent when SSL Card has failed

Varbinds sent in the trap message: [sslCardStatusMsg](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: SSL-CARD-FAILED

sslCardNormal (1.3.6.1.4.1.5951.1.1.0.118)

This trap is sent when SSL Card status returned back to normal

Varbinds sent in the trap message: [sslCardStatusMsg](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: SSL-CARD-NORMAL

warmRestartEvent (1.3.6.1.4.1.5951.1.1.0.119)

This trap is sent when a Warm Restart Event occurred

Varbinds sent in the trap message: [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: WARM-RESTART-EVENT

hardDiskDriveErrors (1.3.6.1.4.1.5951.1.1.0.120)

This trap is sent when Hard Disk Drive Errors are seen on the system

Varbinds sent in the trap message: [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: HARD-DISK-DRIVE-ERRORS

compactFlashErrors (1.3.6.1.4.1.5951.1.1.0.121)

This trap is sent when Compact Flash Errors are seen on the system

Varbinds sent in the trap message: [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: COMPACT-FLASH-ERRORS

callHomeUploadEvent (1.3.6.1.4.1.5951.1.1.0.122)

This trap is sent when an attempt to upload Show Tech Support Archive has been made

Varbinds sent in the trap message: [callHomeUploadEventStatusMsg](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: CALLHOME-UPLOAD-EVENT

rsa1024KeyExThresholdHigh (1.3.6.1.4.1.5951.1.1.0.123)

This trap is sent when RSA 1024 key exchange limit has exceeded the threshold

Varbinds sent in the trap message: [alarmHighThreshold](#), [alarmCurrentValue](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: 1024KEY-EXCHANGE-RATE

rsa1024KeyExThresholdNormal (1.3.6.1.4.1.5951.1.1.0.124)

This trap is sent when RSA 1024 key exchange limit returns back to normal

Varbinds sent in the trap message: [alarmNormalThreshold](#), [alarmCurrentValue](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: 1024KEY-EXCHANGE-RATE

rsa2048KeyExThresholdHigh (1.3.6.1.4.1.5951.1.1.0.125)

This trap is sent when RSA 2048 key exchange rate limit has exceeded the threshold

Varbinds sent in the trap message: [alarmHighThreshold](#), [alarmCurrentValue](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: 2048KEY-EXCHANGE-RATE

rsa2048KeyExThresholdNormal (1.3.6.1.4.1.5951.1.1.0.126)

This trap is sent when RSA 2048 key exchange rate limit returns back to normal

Varbinds sent in the trap message: [alarmNormalThreshold](#), [alarmCurrentValue](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: 2048KEY-EXCHANGE-RATE

rsa4096KeyExThresholdHigh (1.3.6.1.4.1.5951.1.1.0.127)

This trap is sent when RSA 4096 key exchange rate limit has exceeded the threshold

Varbinds sent in the trap message: [alarmHighThreshold](#), [alarmCurrentValue](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: 4096KEY-EXCHANGE-RATE

rsa4096KeyExThresholdNormal (1.3.6.1.4.1.5951.1.1.0.128)

This trap is sent when RSA 4096 key exchange rate limit returns back to normal

Varbinds sent in the trap message: [alarmNormalThreshold](#), [alarmCurrentValue](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: 4096KEY-EXCHANGE-RATE

sslCurSessionInUseHigh (1.3.6.1.4.1.5951.1.1.0.129)

This trap is sent when SSL current session in use has exceeded the threshold

Varbinds sent in the trap message: [alarmHighThreshold](#), [alarmCurrentValue](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: SSL-CUR-SESSION-INUSE

sslCurSessionInUseNormal (1.3.6.1.4.1.5951.1.1.0.130)

This trap is sent when SSL current session in use returns back to normal

Varbinds sent in the trap message: [alarmNormalThreshold](#), [alarmCurrentValue](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: SSL-CUR-SESSION-INUSE

clusterNodeHealth (1.3.6.1.4.1.5951.1.1.0.131)

This trap is sent by all cluster nodes when their health state changes. This trap is also sent when a peer node goes down.

Varbinds sent in the trap message: [clNodeIP](#), [clNodeEffectiveHealth](#), [clNodeHealthReason](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: CLUSTER-NODE-HEALTH

clusterNodeQuorum (1.3.6.1.4.1.5951.1.1.0.132)

This trap indicates whether the node view of cluster has quorum or not.

Varbinds sent in the trap message: [clNodeViewQuorum](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: CLUSTER-NODE-QUORUM

clusterVersionMismatch (1.3.6.1.4.1.5951.1.1.0.133)

This trap is sent when there is a version mismatch among the cluster nodes.

Varbinds sent in the trap message: [clNodeIP](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: CLUSTER-VERSION-MISMATCH

clusterCCOChange (1.3.6.1.4.1.5951.1.1.0.134)

This trap is sent when the Configuration Coordinator of the cluster changes.

Varbinds sent in the trap message: [oldCCOIP](#), [newCCOIP](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: CLUSTER-CCO-CHANGE

clusterOVSCheck (1.3.6.1.4.1.5951.1.1.0.135)

This trap is sent when cluster operational view set(OVS) changes.

Varbinds sent in the trap message: [oldOVS](#), [newOVS](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: CLUSTER-OVS-CHANGE

clusterSyncFailure (1.3.6.1.4.1.5951.1.1.0.136)

This trap is sent by cluster nodes when there is a sync failure.

Varbinds sent in the trap message: [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: CLUSTER-SYNC-FAILURE

clusterPropFailure (1.3.6.1.4.1.5951.1.1.0.137)

This trap is sent when cluster propagation of configurations fails/times out.

Varbinds sent in the trap message: [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: CLUSTER-PROP-FAILURE

stickyPrimary (1.3.6.1.4.1.5951.1.1.0.138)

This trap is sent when max flips are completed and we do not give up primary ownership inspite of route monitor failure.

Varbinds sent in the trap message: [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: HA-STICKY-PRIMARY

inbandProtocolVersionMismatch (1.3.6.1.4.1.5951.1.1.0.139)

This trap is sent when there is inband protocol mismatch between Qosd and BR.

Varbinds sent in the trap message: [qosdVersion](#), [brVersion](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: INBAND-PROTOCOL-VERSION-MISMATCH

sslChipReinit (1.3.6.1.4.1.5951.1.1.0.140)

This trap is sent when a SSL chip reinitialize occurs.

Varbinds sent in the trap message: [sslChipName](#), [sslChipReinitCount](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: SSL-CHIP-REINIT

appfwViolations (1.3.6.1.4.1.5951.1.1.0.141)

This trap indicates that AppFirewall Unknow Content-Type violation occurred.

Varbinds sent in the trap message: [appfwLogMsg](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: APPFW-VIOLATIONS-TYPE

vridStateChange (1.3.6.1.4.1.5951.1.1.0.142)

This trap is sent when the state of VRID changes in ACTIVE/ACTIVE setup

Varbinds sent in the trap message: [vrid](#), [vridBoundVIP](#), [newVridPriority](#), [effectiveVridPriority](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: VRID-STATE-CHANGE

portAllocFailed (1.3.6.1.4.1.5951.1.1.0.143)

This trap is sent on port allocation failure

Varbinds sent in the trap message: [dstip](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: PORT-ALLOC-FAILED

IldpRemTablesChange (1.3.6.1.4.1.5951.1.1.0.144)

This trap is sent on any insert/delete in IldpRemManAddrTable

Varbinds sent in the trap message: [IldpRemLocalPortNum](#), [IldpRemChassisId](#), [IldpRemPortId](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: LLDP-REMOTE-CHANGE

ipv6AddressDuplicated (1.3.6.1.4.1.5951.1.1.0.145)

This trap indicates that ipv6 address got duplicated in the network.

Varbinds sent in the trap message: [nsIPAddressType](#), [nsIPAddress](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: DUPLICATE-IPV6

haVersionMatched (1.3.6.1.4.1.5951.1.1.0.149)

This trap indicates that the mismatched OS version of the netscalers in HA has been corrected.

Varbinds sent in the trap message: [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: HA-VERSION-MISMATCH

haSyncSucceeded (1.3.6.1.4.1.5951.1.1.0.150)

This trap indicates that config synchronization has succeeded on secondary.

Varbinds sent in the trap message: [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: HA-SYNC-FAILURE

haSecondaryStateNormal (1.3.6.1.4.1.5951.1.1.0.151)

This trap indicates that the secondary has come back to normal UP state.

Varbinds sent in the trap message: [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: HA-BAD-SECONDARY-STATE

haHeartbeatsRecvd (1.3.6.1.4.1.5951.1.1.0.152)

This trap indicates that Heartbeats have been received on the specified interface.

Varbinds sent in the trap message: [haNicMonitorSucceeded](#), [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: HA-NO-HEARTBEATS

haPropSuccess (1.3.6.1.4.1.5951.1.1.0.154)

This trap indicates that config propagation has succeeded on secondary after a previous failure.

Varbinds sent in the trap message: [sysIpAddress](#)

To receive this trap, enable the following SNMP alarm: HA-PROP-FAILURE

NetScaler Command Reference

A detailed list of the commands that can be used to configure the NetScaler appliance through the CLI.

- o AAA Commands
- o Application Commands
- o AppFlow Commands
- o Application Firewall Commands
- o AppQoE Commands
- o Audit Commands
- o Authentication Commands
- o Authorization Commands
- o AutoScale Commands
- o Basic Commands
- o Content Accelerator Commands
- o Cache Commands
- o CLI Commands
- o Cluster Commands
- o Compression Commands
- o Cache Redirection Commands
- o Content Switching Commands
- o DB Commands
- o DNS Commands
- o DOS Commands
- o Front End Optimization Commands
- o Filter Commands
- o GSLB Commands
- o High Availability Commands
- o IPsec Commands
- o Load Balancing Commands
- o LLDP Commands
- o Networking Commands
- o NS Commands
- o NTP Commands
- o Policy Commands
- o PQ Commands
- o Protocol Commands
- o QOS Commands
- o Responder Commands
- o Rewrite Commands
- o RISE Commands
- o Router Commands
- o SureConnect Commands
- o SNMP Commands
- o Spillover Commands
- o SSL Commands
- o Stream Commands
- o System Commands
- o Traffic Management Commands
- o Transform Commands
- o Tunnel Commands
- o Utility Commands
- o VPN Commands
- o WebInterface Commands

AAA Commands

The entities on which you can perform NetScaler CLI operations:

- o `aaa`
- o `aaa certParams`
- o `aaa global`
- o `aaa group`
- o `aaa kcdAccount`
- o `aaa ldapParams`
- o `aaa parameter`
- o `aaa preauthenticationaction`
- o `aaa preauthenticationparameter`
- o `aaa preauthenticationpolicy`
- o `aaa radiusParams`
- o `aaa session`
- o `aaa stats`
- o `aaa tacacsParams`
- o `aaa user`

aaa

The following operations can be performed on "aaa":

stat aaa

Display aaa statistics

Synopsys

```
stat aaa [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

Authentication successes (authsucc)

Count of authentication successes.

Authentication failures (authfails)

Count of authentication failures.

HTTP authorization successes (atzhttps)

Count of HTTP connections that succeeded authorization.

HTTP authorization failures (atzhttpf)

Count of HTTP connections that failed authorization.

Non HTTP authorization successes (atznonhttps)

Count of non HTTP connections that succeeded authorization.

Non HTTP authorization failures (atznonhttpf)

Count of non HTTP connections that failed authorization.

Current AAA sessions (totcursess)

Count of current AAA sessions.

Total AAA sessions (totsess)

Count of all AAA sessions.

Timed out AAA sessions (totsessto)

Count of AAA sessions that have timed out.

Current ICAOnly sessions (totcuricasess)

Count of current ICA only sessions.

Current ICAOnly Conn (curicaonlyconn)

Count of current ICA only connections.

Current ICA (Smart Access) Conn (curicaconn)

Count of current ICA connections.

Current TM sessions (curTMses)

Count of current AAATM sessions.

TM sessions (totTMses)

Count of all AAATM sessions.

aaa certParams

The following operations can be performed on "aaa certParams":

[set](#) | [unset](#) | [show](#)

set aaa certParams

Modifies the global configuration settings for certificate policies. The settings that you specify are used for all SSL-VPN virtual servers unless you use authentication policies to create a configuration for a specific SSL-VPN virtual server.

Synopsys

```
set aaa certParams [-userNameField <string>] [-groupNameField <string>] [-defaultAuthenticationGroup <string>]
```

Arguments

userNameField

Client certificate field that contains the username, in the format <field>:<subfield>.

groupNameField

Client certificate field that specifies the group, in the format <field>:<subfield>.

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Maximum value: 64

Example

```
To configure the default certificate parameters:  set aaa certparams -userNameField "Subject"
```

unset aaa certParams

Use this command to remove aaa certParams settings. Refer to the set aaa certParams command for meanings of the arguments.

Synopsys

```
unset aaa certParams [-userNameField] [-groupNameField] [-defaultAuthenticationGroup]
```

show aaa certParams

Displays the current client certificate configuration on the NetScaler appliance.

Synopsys

```
show aaa certParams
```

Outputs

twoFactor

The state of the two-factor authentication.

userNameField

The field in the certificate from which the username will be extracted.

groupNameField

The field in the certificate from which the group will be extracted.

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

aaa global

The following operations can be performed on "aaa global":

[bind](#) | [unbind](#) | [show](#)

bind aaa global

Binds a policy globally.

Synopsis

```
bind aaa global [-policy <string> [-priority <positive_integer>]] [-windowsProfile <string>]
```

Arguments

policy

Name of the policy to bind globally.

priority

Priority to assign to the policy, as an integer. A lower number indicates a higher priority. Policies are evaluated in the order of their priority numbers.

Minimum value: 0

windowsProfile

Name of the negotiate profile to bind globally.

Example

```
bind aaa global -pol poll
```

unbind aaa global

Unbind the policy from the global bind point.

Synopsis

```
unbind aaa global [-policy <string>] [-windowsProfile <string>]
```

Arguments

policy

Name of the policy to be unbound.

windowsProfile

Name of the negotiate profile to be bound.

show aaa global

Displays a list of policies that are currently bound to Global on the NetScaler appliance.

Synopsis

```
show aaa global
```

Outputs

policy

Name of the policy to be unbound.

windowsProfile

Name of the negotiate profile to be bound.

priority

Priority of the bound policy

bindPolicyType

Bound policy type

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

devno**count****stateflag**

aaa group

The following operations can be performed on "aaa group":

add | **rm** | **bind** | **unbind** | **show**

add aaa group

Creates a AAA group and verifies the configuration to ensure that it is correct.

Synopsis

```
add aaa group <groupName>
```

Arguments

groupName

Name for the group. Must begin with a letter, number, or the underscore character (_), and must consist only of letters, numbers, and the hyphen (-), period (.), pound (#), space (), at sign (@), equals (=), colon (:), and underscore characters. Cannot be changed after the group is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my aaa group" or 'my aaa group').

Example

```
add aaa group group_ad
```

rm aaa group

Removes the specified AAA group.

Synopsis

```
rm aaa group <groupName>
```

Arguments

groupName

Name of the group that you are removing.

bind aaa group

Binds the specified AAA group to the specified resource. The resource can be a user, an Intranet IP address or range, a policy, or an Intranet application.

Synopsis

```
bind aaa group <groupName> [-userName <string>] [-policy <string> [-priority <positive_integer>]] [-intranetApplication <string>] [-urlName <string>] [-intranetIP <ip_addr> <netmask>]
```

Arguments

groupName

Name of the group that you are binding.

userName

Bind a AAA group to the specified AAA user.

If the specified user is bound to more than one group, the group expressions are evaluated, upon authorization, to determine the appropriate action.

policy

Bind a policy to the specified AAA group.

priority

Priority to assign to the policy, as an integer. A lower number indicates a higher priority.

Required when binding a group to a policy. Not relevant to any other type of group binding.

Minimum value: 0

intranetApplication

Bind the group to the specified intranet VPN application.

urlName

Bind the group to the specified URL.

intranetIP

Bind the group to the specified IP address or IP block.

Normally you would bind the group to an IP address or range that your users use to access intranet resources.

netmask

Subnet mask specifying an IP-address range to which to bind a AAA group.

Example

To bind an Intranet IP to the group engg: `bind aaa group engg -intranetip 10.102.10.0 255.255.255.0`

unbind aaa group

Unbinds the specified AAA group from the specified resource. The resource can be a user, an intranet IP address or range, a policy, or an intranet application.

Synopsis

```
unbind aaa group <groupName> [-userName <string> ...] [-policy <string>] [-intranetApplication <string>] [-urlName <string>] [-intranetIP <ip_addr> <netmask>]
```

Arguments

groupName

Name of the group that you are unbinding.

userName

Unbind the specified AAA group from the specified AAA user.

policy

Unbind the specified policy from the specified AAA group.

intranetApplication

Unbind the specified group from the specified intranet VPN application.

urlName

Unbind the specified group from the specified URL.

intranetIP

Unbind the specified group from the specified IP address or IP block.

netmask

Subnet mask for the IP range in which the intranet application from which you are unbinding the policy resides.

Required if the intranet application has multiple IP addresses bound to it. Not needed if the intranet application resides on a single IP address.

Example

```
unbind aaa group engg -intranetip 10.102.10.0 255.255.255.0
```

show aaa group

Displays the current configuration of a AAA group.

Synopsys

```
show aaa group [<groupName>] [-loggedIn]
```

Arguments

groupName

Name of the group.

loggedIn

Display only the group members who are currently logged in.

Outputs

userName

The user name.

policy

The policy name.

priority

Priority to assign to the policy, as an integer. A lower number indicates a higher priority.

Required when binding a group to a policy. Not relevant to any other type of group binding.

intranetApplication

Bind the group to the specified intranet VPN application.

urlName

The intranet url

actType

intranetIP

The Intranet IP(s) bound to the group

netmask

The netmask for the Intranet IP

policySubType

stateflag

devno

count

Example

```
> show aaa group engg      GroupName:  engg      Bound AAA users:      UserName:
```

aaa kcdAccount

The following operations can be performed on "aaa kcdAccount":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add aaa kcdAccount

Add a Kerberos constrained delegation account.

Synopsys

```
add aaa kcdAccount <kcdAccount> {-keytab <string>} {-realmStr <string>} {-delegatedUser <string>} {-kcdPassword  
<string>} {-usercert <string>} {-cacert <string>} [-userRealm <string>] [-enterpriseRealm <string>] [-serviceSPN <string>]}
```

Arguments

kcdAccount

The name of the KCD account.

keytab

The path to the keytab file. If specified other parameters in this command need not be given

realmStr

Kerberos Realm.

delegatedUser

Username that can perform kerberos constrained delegation.

kcdPassword

Password for Delegated User.

usercert

SSL Cert (including private key) for Delegated User.

cacert

CA Cert for UserCert or when doing PKINIT backchannel.

userRealm

Realm of the user

enterpriseRealm

Enterprise Realm of the user. This should be given only in certain KDC deployments where KDC expects Enterprise username instead of Principal Name

serviceSPN

Service SPN. When specified, this will be used to fetch kerberos tickets. If not specified, Netscaler will construct SPN using service fqdn

Example

```
add aaa kcdaccount my_kcd_acct -keytab /var/mykcd.keytab add aaa kcdaccount my_kcd_acct -l
```

rm aaa kcdAccount

Remove the KCD account.

Synopsys

rm aaa kcdAccount <kcdAccount>

Arguments

kcdAccount

The KCD account name.

set aaa kcdAccount

Set the KCD account information.

Synopsys

set aaa kcdAccount <kcdAccount> [-keytab <string>] [-realmStr <string>] [-delegatedUser <string>] [-kcdPassword] [-usercert <string>] [-cacert <string>] [-userRealm <string>] [-enterpriseRealm <string>] [-serviceSPN <string>]

Arguments

kcdAccount

The name of the KCD account.

keytab

The path to the keytab file. If specified other parameters in this command need not be given

realmStr

Kerberos Realm.

delegatedUser

Username that can perform kerberos constrained delegation.

kcdPassword

Password for Delegated User.

usercert

SSL Cert (including private key) for Delegated User.

cacert

CA Cert for UserCert or when doing PKINIT backchannel.

userRealm

Realm of the user

enterpriseRealm

Enterprise Realm of the user. This should be given only in certain KDC deployments where KDC expects Enterprise username instead of Principal Name

serviceSPN

Service SPN. When specified, this will be used to fetch kerberos tickets. If not specified, Netscaler will construct SPN using service fqdn

Example

set aaa kcdaccount my_kcd_acct -keytab /var/hiskcd.keytab The above command sets the keyt:

unset aaa kcdAccount

Unset the KCD account information..Refer to the set aaa kcdAccount command for meanings of the arguments.

Synopsys

```
unset aaa kcdAccount <kcdAccount> [-usercert] [-cacert] [-userRealm] [-enterpriseRealm] [-serviceSPN]
```

show aaa kcdAccount

Display KCD accounts.

Synopsys

```
show aaa kcdAccount [<kcdAccount>]
```

Arguments

kcdAccount

The KCD account name.

Outputs

keytab

The path to the keytab file. If specified other parameters in this command need not be given

principle

SPN extracted from keytab file.

kcdSPN

Host SPN extracted from keytab file.

realmStr

Kerberos Realm.

delegatedUser

Username that can perform kerberos constrained delegation.

kcdPassword

Password for Delegated User.

usercert

SSL Cert (including private key) for Delegated User.

cacert

CA Cert for UserCert or when doing PKINIT backchannel.

userRealm

Realm of the user

enterpriseRealm

Enterprise Realm of the user. This should be given only in certain KDC deployments where KDC expects Enterprise username instead of Principal Name

serviceSPN

Service SPN. When specified, this will be used to fetch kerberos tickets. If not specified, Netscaler will construct SPN using service fqdn

stateflag

devno

count

Example

```
Example > show aaa kcdaccount my_kcd_acct
```

```
KcdAccount: my_kcd_acct
```

```
Keytab
```

aaa ldapParams

The following operations can be performed on "aaa ldapParams":

[set](#) | [unset](#) | [show](#)

set aaa ldapParams

Modifies the global configuration settings for the LDAP server. The settings that you specify are used for all SSL-VPN virtual servers unless you use authentication policies to create a configuration for a specific SSL-VPN virtual server.

Synopsys

```
set aaa ldapParams [-serverIP <ip_addr|ipv6_addr|*>] [-serverPort <port>] [-authTimeout <positive_integer>] [-ldapBase <string>] [-ldapBindDn <string>] {-ldapBindDnPassword } [-ldapLoginName <string>] [-searchFilter <string>] [-groupAttrName <string>] [-subAttributeName <string>] [-secType <secType>] [-svrType ( AD | NDS )] [-ssoNameAttribute <string>] [-passwdChange ( ENABLED | DISABLED )] [-nestedGroupExtraction ( ON | OFF )] [-maxNestingLevel <positive_integer>] [-groupNameIdentifier <string>] [-groupSearchAttribute <string>] [-groupSearchSubAttribute <string>] [-groupSearchFilter <string>] [-defaultAuthenticationGroup <string>]
```

Arguments

serverIP

IP address of your LDAP server.

serverPort

Port number on which the LDAP server listens for connections.

Default value: 389

Minimum value: 1

authTimeout

Maximum number of seconds that the NetScaler appliance waits for a response from the LDAP server.

Default value: 3

Minimum value: 1

ldapBase

Base (the server and location) from which LDAP search commands should start.

If the LDAP server is running locally, the default value of base is dc=netscaler, dc=com.

ldapBindDn

Complete distinguished name (DN) string used for binding to the LDAP server.

ldapBindDnPassword

Password for binding to the LDAP server.

ldapLoginName

Name attribute that the NetScaler appliance uses to query the external LDAP server or an Active Directory.

searchFilter

String to be combined with the default LDAP user search string to form the value to use when executing an LDAP search.

For example, the following values:

vpnallowed=true,

ldaploginame=""samaccount""

when combined with the user-supplied username ""bob"", yield the following LDAP search string:

""(&(vpnallowed=true)(samaccount=bob)""

groupAttrName

Attribute name used for group extraction from the LDAP server.

subAttributeName

Subattribute name used for group extraction from the LDAP server.

secType

Type of security used for communications between the NetScaler appliance and the LDAP server. For the PLAINTEXT setting, no encryption is required.

Possible values: PLAINTEXT, TLS, SSL

Default value: PLAINTEXT

svrType

The type of LDAP server.

Possible values: AD, NDS

Default value: AAA_LDAP_SERVER_TYPE_DEFAULT

ssoNameAttribute

Attribute used by the NetScaler appliance to query an external LDAP server or Active Directory for an alternative username.

This alternative username is then used for single sign-on (SSO).

passwdChange

Accept password change requests.

Possible values: ENABLED, DISABLED

Default value: DISABLED

nestedGroupExtraction

Queries the external LDAP server to determine whether the specified group belongs to another group.

Possible values: ON, OFF

Default value: OFF

maxNestingLevel

Number of levels up to which the system can query nested LDAP groups.

Default value: 2

Minimum value: 2

groupNameIdentifier

LDAP-group attribute that uniquely identifies the group. No two groups on one LDAP server can have the same group name identifier.

groupSearchAttribute

LDAP-group attribute that designates the parent group of the specified group. Use this attribute to search for a group's parent group.

groupSearchSubAttribute

LDAP-group subattribute that designates the parent group of the specified group. Use this attribute to search for a group's parent group.

groupSearchFilter

Search-expression that can be specified for sending group-search requests to the LDAP server.

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Maximum value: 64

Example

To configure authentication in the LDAP server running at 192.40.1.2: `set aaa ldapparams`

unset aaa ldapParams

Use this command to remove aaa ldapParams settings. Refer to the set aaa ldapParams command for meanings of the arguments.

Synopsis

`unset aaa ldapParams [-serverIP] [-serverPort] [-authTimeout] [-ldapBase] [-ldapBindDn] [-ldapBindDnPassword] [-ldapLoginName] [-searchFilter] [-groupAttrName] [-subAttributeName] [-secType] [-svrType] [-ssoNameAttribute] [-passwdChange] [-nestedGroupExtraction] [-maxNestingLevel] [-groupNameIdentifier] [-groupSearchAttribute] [-groupSearchSubAttribute] [-groupSearchFilter] [-defaultAuthenticationGroup]`

show aaa ldapParams

Displays the current LDAP configuration on the NetScaler appliance.

Synopsis

`show aaa ldapParams`

Outputs

serverIP

The IP address of the LDAP server.

serverPort

Port number on which the LDAP server listens for connections.

authTimeout

Maximum number of seconds that the NetScaler appliance waits for a response from the LDAP server.

ldapBindDn

The full distinguished name used to bind to the LDAP server.

ldapLoginName

The name attribute used by the system to query the external LDAP server, or an Active Directory.

ldapBase

The base or node where the ldapsearch should start. If the LDAP server is running locally, the default value of base is `dc=netScaler, dc=com`.

secType

The communication type between the system and the LDAP server.

svrType

LDAP server.

ssoNameAttribute

The attribute used by the system to query the external LDAP server, or an Active Directory, for an alternate username to be used in Single Sign-On.

searchFilter

The String to be combined with the default LDAP user search string to form the value. For example, `vpnallowed=true` with `ldaploginame "samaccount"` and the user-supplied username "bob" would yield the LDAP search string `"(&(vpnallowed=true)(samaccount=bob))"`.

groupAttrName

The Attribute name for group extraction from the LDAP server.

subAttributeName

Subattribute name used for group extraction from the LDAP server.

groupAuthName

To associate AAA users with an AAA group, use the command

`"bind AAA group ... -username ..."`.

You can bind different policies to each AAA group. Use the command

`"bind AAA group ... -policy ..."`

passwdChange

Accept password change requests.

nestedGroupExtraction

Queries the external LDAP server to determine whether the specified group belongs to another group.

maxNestingLevel

Number of levels up to which the system can query nested LDAP groups.

groupNameIdentifier

LDAP-group attribute that uniquely identifies the group. No two groups on one LDAP server can have the same group name identifier.

groupSearchAttribute

LDAP-group attribute that designates the parent group of the specified group. Use this attribute to search for a group's parent group.

groupSearchSubAttribute

LDAP-group subattribute that designates the parent group of the specified group. Use this attribute to search for a group's parent group.

groupSearchFilter

Search-expression that can be specified for sending group-search requests to the LDAP server.

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Example

```
> show aaa ldapparams Configured LDAP parameters   Server IP: 127.0.0.1   Port: 389   Time
```

aaa parameter

The following operations can be performed on "aaa parameter":

[set](#) | [unset](#) | [show](#)

set aaa parameter

Sets the global AAA configuration. Any configuration settings made at this level overrides configuration settings for the authentication server.

Synopsys

```
set aaa parameter [-enableStaticPageCaching ( YES | NO )] [-enableEnhancedAuthFeedback ( YES | NO )] [-defaultAuthType <defaultAuthType>] [-maxAAAUUsers <positive_integer>] [-maxLoginAttempts <positive_integer>] [-failedLoginTimeout <mins>]] [-aaadnatlp <ip_addr|*>] [-enableSessionStickiness ( YES | NO )]
```

Arguments

enableStaticPageCaching

The default state of VPN Static Page caching. If nothing is specified, the default value is set to YES.

Possible values: YES, NO

Default value: YES

enableEnhancedAuthFeedback

Enhanced auth feedback provides more information to the end user about the reason for an authentication failure. The default value is set to NO.

Possible values: YES, NO

Default value: NO

defaultAuthType

The default authentication server type.

Possible values: LOCAL, LDAP, RADIUS, TACACS, CERT

Default value: LOCAL

maxAAAUUsers

Maximum number of concurrent users allowed to log on to VPN simultaneously.

Minimum value: 1

maxLoginAttempts

Maximum Number of login Attempts

Minimum value: 1

failedLoginTimeout

Number of minutes an account will be locked if user exceeds maximum permissible attempts

Minimum value: 1

aaadnatlp

Source IP address to use for traffic that is sent to the authentication server.

enableSessionStickiness

Enables/Disables stickiness to authentication servers

Possible values: YES, NO

Default value: NO

Example

```
set aaa parameter -defaultAuthType RADIUS -maxAAUSers 100
```

unset aaa parameter

Resets the global AAA parameter settings on the NetScaler appliance. Attributes for which a default value is available revert to their default values. See the set aaa parameter command for descriptions of the parameters..Refer to the set aaa parameter command for meanings of the arguments.

Synopsys

```
unset aaa parameter [-enableStaticPageCaching] [-enableEnhancedAuthFeedback] [-defaultAuthType] [-maxAAUSers] [-aaadnatlp] [-maxLoginAttempts] [-enableSessionStickiness]
```

show aaa parameter

Displays the current AAA global configuration.

Synopsys

```
show aaa parameter
```

Outputs

enableStaticPageCaching

Indicates if static page caching is enabled or not.

enableEnhancedAuthFeedback

Indicates whether enhanced auth feedback is enabled or not.

defaultAuthType

The default authentication server type.

maxAAUSers

The maximum number of concurrent users allowed to log into the system at any time.

aaadnatlp

The natlp to be used for the AAA traffic

maxLoginAttempts

Maximum Number of login Attempts

failedLoginTimeout

Number of minutes an account will be locked if user exceeds maximum permissible attempts

enableSessionStickiness

Enables/Disables stickiness to authentication servers

Example

```
> show aaa parameter Configured AAA parameters           DefaultAuthType: LDAP      MaxAAUser:
```


aaa preauthenticationaction

The following operations can be performed on "aaa preauthenticationaction":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add aaa preauthenticationaction

Adds an action (profile) for endpoint analysis (EPA) clients before authentication.

Synopsis

```
add aaa preauthenticationaction <name> [<preauthenticationaction>] [-killProcess <string>] [-deletefiles <string>]
```

Arguments

name

Name for the preauthentication action. Must begin with a letter, number, or the underscore character (_), and must consist only of letters, numbers, and the hyphen (-), period (.), pound (#), space (), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after preauthentication action is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my aaa action" or 'my aaa action').

preauthenticationaction

Allow or deny logon after endpoint analysis (EPA) results.

Possible values: ALLOW, DENY

killProcess

String specifying the name of a process to be terminated by the endpoint analysis (EPA) tool.

deletefiles

String specifying the path(s) and name(s) of the files to be deleted by the endpoint analysis (EPA) tool.

rm aaa preauthenticationaction

Removes a preauthentication action. NOTE: A preauthentication action cannot be removed if it is bound to a policy.

Synopsis

```
rm aaa preauthenticationaction <name>
```

Arguments

name

Name of the preauthentication action to remove.

set aaa preauthenticationaction

Modifies an existing preauthentication action (profile).

Synopsis

```
set aaa preauthenticationaction <name> [<preauthenticationaction>] [-killProcess <string>] [-deletefiles <string>]
```

Arguments

name

Name of the preauthentication action to modify.

preauthenticationaction

Allow or deny logon after endpoint analysis (EPA) results.

Possible values: ALLOW, DENY

killProcess

String specifying the name of a process to be terminated by the endpoint analysis (EPA) tool.

deletefiles

String specifying the path(s) and name(s) of the files to be deleted by the endpoint analysis (EPA) tool.

unset aaa preauthenticationaction

Use this command to remove aaa preauthenticationaction settings. Refer to the set aaa preauthenticationaction command for meanings of the arguments.

Synopsis

```
unset aaa preauthenticationaction <name> [-killProcess] [-deletefiles]
```

show aaa preauthenticationaction

Displays details of the specified preauthentication action.

Synopsis

```
show aaa preauthenticationaction [<name>]
```

Arguments

name

Name of the preauthentication action.

Outputs

preauthenticationaction

Allow or deny logon after endpoint analysis (EPA) results.

killProcess

String specifying the name of a process to be terminated by the endpoint analysis (EPA) tool.

deletefiles

String specifying the path(s) and name(s) of the files to be deleted by the endpoint analysis (EPA) tool.

stateflag**builtin**

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

devno**count**

aaa preauthenticationparameter

The following operations can be performed on "aaa preauthenticationparameter":

[set](#) | [unset](#) | [show](#)

set aaa preauthenticationparameter

Configures the default end point analysis (EPA) parameters that are applied before authentication.

Synopsys

```
set aaa preauthenticationparameter [-preauthenticationaction ( ALLOW | DENY )] [-rule <expression>] [-killProcess <string>] [-deletefiles <string>]
```

Arguments

preauthenticationaction

Deny or allow login on the basis of end point analysis results.

Possible values: ALLOW, DENY

rule

Name of the NetScaler named rule, or a default syntax expression, to be evaluated by the EPA tool.

killProcess

String specifying the name of a process to be terminated by the EPA tool.

deletefiles

String specifying the path(s) to and name(s) of the files to be deleted by the EPA tool, as a string of between 1 and 1023 characters.

unset aaa preauthenticationparameter

Resets the default end point analysis(EPA) configuration settings on the NetScaler appliance. Attributes for which a default value is available revert to their default values. See the set aaa preauthenticationparameter command for descriptions of the parameters..Refer to the set aaa preauthenticationparameter command for meanings of the arguments.

Synopsys

```
unset aaa preauthenticationparameter [-rule] [-preauthenticationaction] [-killProcess] [-deletefiles]
```

show aaa preauthenticationparameter

Displays the current preauthentication configuration.

Synopsys

```
show aaa preauthenticationparameter
```

Outputs

preauthenticationaction

Deny or allow login after End point analysis results.

rule

Name of the NetScaler named rule, or a default syntax expression, to be evaluated by the EPA tool.

killProcess

Processes to be killed by EPA tool.

deletefiles

Files to be deleted by EPA tool.

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

aaa preauthenticationpolicy

The following operations can be performed on "aaa preauthenticationpolicy":

[add](#) | [rm](#) | [set](#) | [show](#)

add aaa preauthenticationpolicy

Adds a preauthentication policy. The policy defines expressions to be evaluated by the endpoint analysis (EPA) tool.

Synopsis

add aaa preauthenticationpolicy <name> <rule> [<reqAction>]

Arguments

name

Name for the preauthentication policy. Must begin with a letter, number, or the underscore character (_), and must consist only of letters, numbers, and the hyphen (-), period (.), pound (#), space (), at sign (@), equals (=), colon (:), and underscore characters. Cannot be changed after the preauthentication policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my policy" or 'my policy').

rule

Name of the NetScaler named rule, or a default syntax expression, defining connections that match the policy.

reqAction

Name of the action that the policy is to invoke when a connection matches the policy.

rm aaa preauthenticationpolicy

Removes the specified preauthentication policy.

Synopsis

rm aaa preauthenticationpolicy <name>

Arguments

name

Name of the preauthentication policy to remove.

set aaa preauthenticationpolicy

Modifies the Request Action of a preauthentication policy.

Synopsis

set aaa preauthenticationpolicy <name> [-rule <expression>] [-reqAction <string>]

Arguments

name

Name of the preauthentication policy to modify.

rule

The new rule to be associated with the policy.

reqAction

Name of the action that the policy is to invoke when a connection matches the policy.

show aaa preauthenticationpolicy

Displays the properties of either the specified preauthentication policy or (if none is specified) a list of all configured preauthentication policies.

Synopsys

show aaa preauthenticationpolicy [<name>]

Arguments

name

Name of the preauthentication policy whose properties you want to view.

Outputs

rule

The new rule associated with the policy.

reqAction

The Pre-authentication action associated with the policy.

hits

No of hits.

boundTo

The entity name to which policy is bound

activePolicy**priority****bindPolicyType****policyType****builtin**

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

devno**count****stateflag**

aaa radiusParams

The following operations can be performed on "aaa radiusParams":

[set](#) | [unset](#) | [show](#)

set aaa radiusParams

Modifies the global configuration settings for the RADIUS server. The settings that you specify are used for all SSL-VPN virtual servers unless you use authentication policies to create a configuration for a specific SSL-VPN virtual server.

Synopsys

```
set aaa radiusParams [-serverIP <ip_addr|ipv6_addr|*>] [-serverPort <port>] [-authTimeout <positive_integer>] {-radKey } [-radNASip ( ENABLED | DISABLED )] [-radNASid <string>] [-radVendorID <positive_integer>] [-radAttributeType <positive_integer>] [-radGroupsPrefix <string>] [-radGroupSeparator <string>] [-passEncoding <passEncoding>] [-ipVendorID <positive_integer>] [-ipAttributeType <positive_integer>] [-accounting ( ON | OFF )] [-pwdVendorID <positive_integer>] [-pwdAttributeType <positive_integer>] [-defaultAuthenticationGroup <string>] [-callingstationid ( ENABLED | DISABLED )]
```

Arguments

serverIP

IP address of your RADIUS server.

serverPort

Port number on which the RADIUS server listens for connections.

Default value: 1812

Minimum value: 1

authTimeout

Maximum number of seconds that the NetScaler appliance waits for a response from the RADIUS server.

Default value: 3

Minimum value: 1

radKey

The key shared between the RADIUS server and clients.

Required for allowing the NetScaler appliance to communicate with the RADIUS server.

radNASip

Send the NetScaler IP (NSIP) address to the RADIUS server as the Network Access Server IP (NASIP) part of the Radius protocol.

Possible values: ENABLED, DISABLED

radNASid

Send the Network Access Server ID (NASID) for your NetScaler appliance to the RADIUS server as the nasid part of the Radius protocol.

radVendorID

Vendor ID for RADIUS group extraction.

Minimum value: 1

radAttributeType

Attribute type for RADIUS group extraction.

Minimum value: 1

radGroupsPrefix

Prefix string that precedes group names within a RADIUS attribute for RADIUS group extraction.

radGroupSeparator

Group separator string that delimits group names within a RADIUS attribute for RADIUS group extraction.

passEncoding

Enable password encoding in RADIUS packets that the NetScaler appliance sends to the RADIUS server.

Possible values: pap, chap, mschapv1, mschapv2

Default value: pap

ipVendorID

Vendor ID attribute in the RADIUS response.

If the attribute is not vendor-encoded, it is set to 0.

Minimum value: 0

ipAttributeType

IP attribute type in the RADIUS response.

Minimum value: 1

accounting

Configure the RADIUS server state to accept or refuse accounting messages.

Possible values: ON, OFF

pwdVendorID

Vendor ID of the password in the RADIUS response. Used to extract the user password.

Minimum value: 1

pwdAttributeType

Attribute type of the Vendor ID in the RADIUS response.

Minimum value: 1

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Maximum value: 64

callingstationid

Send Calling-Station-ID of the client to the RADIUS server. IP Address of the client is sent as its Calling-Station-ID.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

To configure the default RADIUS parameters: `set aaa radiusparams -serverip 192.30.1.2 -r`

unset aaa radiusParams

Use this command to remove aaa radiusParams settings. Refer to the set aaa radiusParams command for meanings of the arguments.

Synopsys

```
unset aaa radiusParams [-serverIP] [-serverPort] [-authTimeout] [-radNASip] [-radNASid] [-radVendorID] [-radAttributeType] [-radGroupsPrefix] [-radGroupSeparator] [-passEncoding] [-ipVendorID] [-ipAttributeType] [-accounting] [-pwdVendorID] [-pwdAttributeType] [-defaultAuthenticationGroup] [-callingstationid]
```

show aaa radiusParams

Displays the current RADIUS configuration on the NetScaler appliance.

Synopsys

```
show aaa radiusParams
```

Outputs

serverIP

IP address of your RADIUS server.

serverPort

Port number on which the RADIUS server listens for connections.

radKey

The key shared between the client and the server.

groupAuthName

To associate AAA users with an AAA group, use the command

```
"bind AAA group ... -username ...".
```

You can bind different policies to each AAA group. Use the command

```
"bind AAA group ... -policy ..."
```

authTimeout

Maximum number of seconds that the NetScaler appliance waits for a response from the RADIUS server.

radNASip

The option to send the NetScaler's IP address (NSIP) as the "nasip" (Network Access Server IP) part of the Radius protocol to the server.

radNASid

The nasid (Network Access Server ID). If configured, this string will be sent to the RADIUS server as the "nasid" as part of the Radius protocol.

IPAddress

IP Address.

radVendorID

Vendor ID for RADIUS group extraction.

radAttributeType

Attribute type for RADIUS group extraction.

radGroupsPrefix

Prefix string that precedes group names within a RADIUS attribute for RADIUS group extraction.

radGroupSeparator

Group separator string that delimits group names within a RADIUS attribute for RADIUS group extraction.

passEncoding

Enable password encoding in RADIUS packets that the NetScaler appliance sends to the RADIUS server.

ipVendorID

Vendor ID attribute in the RADIUS response.

If the attribute is not vendor-encoded, it is set to 0.

ipAttributeType

IP attribute type in the RADIUS response.

accounting

The state of the Radius server that will receive accounting messages.

pwdVendorID

Vendor ID of the password in the RADIUS response. Used to extract the user password.

pwdAttributeType

Attribute type of the Vendor ID in the RADIUS response.

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

callingstationid

Send Calling-Station-ID of the client to the RADIUS server. IP Address of the client is sent as its Calling-Station-ID.

Example

```
> show aaa radiusparams Configured RADIUS parameters          Server IP: 127.0.0.2          Po:
```

aaa session

The following operations can be performed on "aaa session":

[show](#) | [kill](#)

show aaa session

Displays all AAA-TM/VPN connections that are bound to the specified user, group, IP address, or IP range.

Synopsys

```
show aaa session [-userName <string>] [-groupName <string>] [-intranetIP <ip_addr|*> [<netmask>]]
```

Arguments

userName

Name of the AAA user.

groupName

Name of the AAA group.

intranetIP

IP address or the first address in the intranet IP range.

netmask

Subnet mask for the intranet IP range.

Outputs

publicIP

Client's public IP address

publicPort

Client's public port

IPAddress

NetScaler's IP address

port

NetScaler's port

privateIP

Client's private/mapped IP address

privatePort

Client's private/mapped port

destIP

Destination IP address

destPort

Destination port

intranetIP

Specifies the Intranet IP

peld

Core id of the session owner

stateflag

devno

count

Example

```
> show aaa connection          ClintIp (ClientPort)    ->  ServerIp(ServerPort)    ----
```

kill aaa session

Terminates the specified AAA-TM/VPN session.

Synopsys

```
kill aaa session [-userName <string>] [-groupName <string>] [-intranetIP <ip_addr|*> [<netmask>]] [-all]
```

Arguments

userName

Terminate AAA-TM/VPN sessions that belong to the specified user.

groupName

Terminate AAA-TM/VPN sessions that belong to any user that is a member of the specified group.

intranetIP

Terminate AAA-TM/VPN sessions that are associated with the specified intranet IP address or with an address in the range specified by the address and subnet mask.

netmask

When terminating AAA-TM/VPN sessions that are associated with an IP address range, the subnet mask defining the range.

all

Terminate all active AAA-TM/VPN sessions.

Example

```
kill aaa session -user joe
```

aaa stats

The following operations can be performed on "aaa stats":

show aaa stats

show aaa stats is an alias for stat aaa

Synopsys

show aaa stats - alias for 'stat aaa'

aaa tacacsParams

The following operations can be performed on "aaa tacacsParams":

[set](#) | [unset](#) | [show](#)

set aaa tacacsParams

Modifies the global configuration settings for the TACACS+ server. The settings that you specify are used for all SSL-VPN virtual servers unless you use authentication policies to create a configuration for a specific SSL-VPN virtual server.

Synopsis

```
set aaa tacacsParams [-serverIP <ip_addr|ipv6_addr*>] [-serverPort <port>] [-authTimeout <positive_integer>] {-tacacsSecret } [-authorization ( ON | OFF )] [-accounting ( ON | OFF )] [-auditFailedCmds ( ON | OFF )] [-defaultAuthenticationGroup <string>]
```

Arguments

serverIP

IP address of your TACACS+ server.

serverPort

Port number on which the TACACS+ server listens for connections.

Default value: 49

Minimum value: 1

authTimeout

Maximum number of seconds that the NetScaler appliance waits for a response from the TACACS+ server.

Default value: 3

Minimum value: 1

tacacsSecret

Key shared between the TACACS+ server and clients. Required for allowing the NetScaler appliance to communicate with the TACACS+ server.

authorization

Use streaming authorization on the TACACS+ server.

Possible values: ON, OFF

accounting

Send accounting messages to the TACACS+ server.

Possible values: ON, OFF

auditFailedCmds

The option for sending accounting messages to the TACACS+ server.

Possible values: ON, OFF

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Maximum value: 64

Example

To configure a TACACS+ server running at 192.168.1.20 `set aaa tacacsparams -serverip 192`

unset aaa tacacsParams

Use this command to remove aaa tacacsParams settings. Refer to the set aaa tacacsParams command for meanings of the arguments.

Synopsys

`unset aaa tacacsParams [-serverIP] [-serverPort] [-authTimeout] [-tacacsSecret] [-authorization] [-accounting] [-auditFailedCmds] [-defaultAuthenticationGroup]`

show aaa tacacsParams

Displays the NetScaler appliance's current AAA TACACS+ configuration.

Synopsys

`show aaa tacacsParams`

Outputs

serverIP

IP address of your TACACS+ server.

serverPort

Port number on which the TACACS+ server listens for connections.

authTimeout

Maximum number of seconds that the NetScaler appliance waits for a response from the TACACS+ server.

tacacsSecret

The key shared between the client and the server.

authorization

The option for the streaming authorization for TACACS+ server.

accounting

The option to send accounting messages to TACACS+ server.

auditFailedCmds

The option to send accounting messages to TACACS+ server.

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Example

`> sh aaa tacacsparams` Configured TACACS parameter Server IP: 192.168.1.20 Port: 41

aaa user

The following operations can be performed on "aaa user":

add | **rm** | **set** | **bind** | **unbind** | **show** | **unlock**

add aaa user

Adds a local AAA user account and verifies the configuration to ensure that it is correct.

Synopsys

```
add aaa user <userName> {-password }
```

Arguments

userName

Name for the user. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.), pound (#), space (), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after the user is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or

single quotation marks (for example, "my aaa user" or 'my aaa user').

password

Password with which the user logs on. Required for any user account that does not exist on an external authentication server.

If you are not using an external authentication server, all user accounts must have a password. If you are using an external authentication server, you must provide a password for local user accounts that do not exist on the authentication server.

Example

```
add aaa user johndoe -password abcd add aaa user johndoe -password The above example adds
```

rm aaa user

Removes a local AAA user account and the associated configuration.

Synopsys

```
rm aaa user <userName>
```

Arguments

userName

Name of the AAA user account to remove.

set aaa user

Configures the password for an existing local AAA user account. This command prompts you for a new password.

NOTE: AAA does not request confirmation of the new password, so you might want to test the new password before sending it to the user.

Synopsys

set aaa user <userName>

Arguments

userName

Name of the local AAA user account.

password

Password with which the user logs on. Required for any user account that does not exist on an external authentication server.

If you are not using an external authentication server, all user accounts must have a password. If you are using an external authentication server, you must provide a password for local user accounts that do not exist on the authentication server.

Example

```
set aaa user johndoe password abcd
```

The above command sets the password for johndoe to abcd

bind aaa user

Binds a policy to the specified user account.

Synopsys

```
bind aaa user <userName> [-policy <string> [-priority <positive_integer>]] [-intranetApplication <string>] [-urlName <string>] [-intranetIP <ip_addr> [<netmask>]]
```

Arguments

userName

User account to which to bind the policy.

policy

Name for the policy that you are creating. Must begin with a letter, number, or the underscore character (_), and must consist only of letters, numbers, and the hyphen (-), period (.), pound (#), space (), at sign (@), equals (=), colon (:), and underscore characters. Cannot be changed after the policy is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my policy" or "my policy").

priority

Integer specifying the priority of the policy. A lower number indicates a higher priority. Policies are evaluated in the order of their priority numbers.

Minimum value: 0

intranetApplication

Name of the intranet VPN application to which the policy applies.

urlName

URL of the intranet application to which you are binding the policy.

intranetIP

IP address of the intranet application to which you are binding the policy.

netmask

Subnet mask for the IP range in which the intranet application to which you are binding the policy resides.

Required if the intranet application has multiple IP addresses bound to

it. Not needed if the intranet application resides on a single IP address.

Example

```
To bind intranetip to the user joe: bind aaa user joe -intranetip 10.102.1.123
```

unbind aaa user

Unbinds a policy from the specified user account.

Synopsys

```
unbind aaa user <userName> [-policy <string>] [-intranetApplication <string>] [-urlName <string>] [-intranetIP <ip_addr> [<netmask>]]
```

Arguments

userName

Name of the user account from which to unbind the policy.

policy

Name of the policy to unbind.

intranetApplication

Name of the intranet VPN application from which you are unbinding the policy.

urlName

URL of the intranet application from which you are unbinding the policy.

intranetIP

Intranet IP address of the application from which you are unbinding the policy.

netmask

Subnet mask for the IP range in which the intranet application from which you are unbinding the policy resides.

Required if the intranet application has multiple IP addresses bound to

it. Not needed if the intranet application resides on a single IP address.

Example

```
unbind AAA user joe -intranetip 10.102.1.123
```

show aaa user

Displays the current configuration of a AAA user account.

Synopsys

```
show aaa user [<userName>] [-loggedIn]
```

Arguments

userName

Name of the user who has the account.

loggedIn

Show whether the user is logged in or not.

Outputs

groupName

The group name

policy

The policy Name.

priority

The priority of the policy.

intranetApplication

Name of the intranet VPN application to which the policy applies.

urlName

The intranet url.

actType

intranetIP

The Intranet IP bound to the user

netmask

The netmask for the Intranet IP

policySubType

stateflag

password

Password with which the user logs on. Required for any user account that does not exist on an external authentication server.

If you are not using an external authentication server, all user accounts must have a password. If you are using an external authentication server, you must provide a password for local user accounts that do not exist on the authentication server.

devno

count

Example

```
Example > show aaa user joe           UserName: joe           IntranetIP: 10.102.1.123
```

unlock aaa user

Unlocks a AAA user account which has been locked earlier for exceeding login attempts.

Synopsys

unlock aaa user <userName>

Arguments

userName

Name of the AAA user account to unlock.

Application Commands

The entities on which you can perform NetScaler CLI operations:

- `application`

application

The following operations can be performed on "application":

import | **export** | **rm**

import application

Imports application configuration information from an AppExpert application template file. You can specify a deployment file along with the template file. A template file contains application and variable definitions. A deployment file contains information about the services, service groups, endpoints, and variables that were in the AppExpert application configuration at the time the template file was created. Before you use template and deployment files, make sure that they are present in the /nsconfig/nstemplates/applications/ and /nsconfig/nstemplates/applications/deployment_files directories, respectively. You can transfer the files from your local drive to those directories on the NetScaler appliance by using either FTP or the NetScaler configuration utility. In the configuration utility, you can also import the files and create the application by using a single wizard (AppExpert > Applications > Import > AppExpert Template Wizard).

Synopsys

```
import application <apptemplateFilename> [-appname <string>] [-deploymentFilename <input_filename>]
```

Arguments

apptemplateFilename

Name of the AppExpert application template file.

appname

Name to assign to the application on the NetScaler appliance. If you do not provide a name, the appliance assigns the application the name of the template file.

deploymentFilename

Name of the deployment file.

Example

```
import app application sampleapp -apptemplatefilename sampleapp.xml -deploymentfilename d
```

export application

Exports application configuration information to an AppExpert application template file. A deployment file is created along with the template file. The template file contains application and variable definitions. The deployment file contains information about the services, service groups, endpoints, and variables that are in the AppExpert application configuration. The template and deployment files are exported to the /nsconfig/nstemplates/applications/ and /nsconfig/nstemplates/applications/deployment_files directories, respectively. If you use the configuration utility, you can also export an application to your local hard drive.

Synopsys

```
export application <appname> [-apptemplateFilename <input_filename>] [-deploymentFilename <input_filename>]
```

Arguments

appname

Name of the AppExpert application whose configuration you want to export to a template file.

apptemplateFilename

Name with which to save the template file. If you do not specify a name, the template file is saved with the name of the application.

deploymentFilename

Name with which to save the deployment file. If you do not specify a name, a string consisting of an underscore and ?deployment? (_deployment) is automatically appended to the name of the template file to create the name of the deployment file.

rm application

Remove application configuration information from a netscaler device. You can specify an application name as input. All the configuration belonging to the specified application will be removed from the device.

Synopsis

```
rm application <appname>
```

Arguments

appname

Name of the AppExpert application whose configuration you want to remove from the Netscaler appliance.

AppFlow Commands

The entities on which you can perform NetScaler CLI operations:

- o appflow
- o appflow action
- o appflow collector
- o appflow global
- o appflow param
- o appflow policy
- o appflow policylabel

appflow

The following operations can be performed on "appflow":

stat appflow

Display AppFlow statistics.

Synopsys

```
stat appflow [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

AppFlow octets transmitted (aflwOcts)

The total number of AppFlow (IPFIX) octets that the NetScaler has transmitted.

AppFlow flows transmitted (aflwFlws)

The total number of AppFlow (IPFIX) flows that the NetScaler has transmitted.

AppFlow messages transmitted (aflwMsgs)

The total number of AppFlow (IPFIX) messages that the NetScaler has transmitted.

Octets ignored for AppFlow (aflwIgnOct)

The total number of octets that the NetScaler has ignored for AppFlow (IPFIX).

Packets ignored for AppFlow (aflwIgnPkts)

The total number of packets that the NetScaler has ignored for AppFlow (IPFIX).

AppFlow octets not transmitted (aflwNoTxOcts)

The total number of AppFlow (IPFIX) octets that the NetScaler has not transmitted.

AppFlow flows not transmitted (aflwNoTxFlows)

The total number of AppFlow (IPFIX) flows that the NetScaler has not transmitted.

AppFlow packets not transmitted (aflwNoTxPkts)

The total number of AppFlow (IPFIX) packets that the NetScaler has not transmitted.

appflow action

The following operations can be performed on "appflow action":

add | **rm** | **set** | **unset** | **rename** | **show**

add appflow action

Creates an AppFlow action. The action can be associated with an AppFlow policy by using the add appflow policy command.

Synopsis

```
add appflow action <name> -collectors <string> ... [-clientSideMeasurements ( ENABLED | DISABLED )] [-comment <string>]
```

Arguments

name

Name for the action. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my appflow action" or 'my appflow action').

collectors

Name(s) of collector(s) to be associated with the AppFlow action.

clientSideMeasurements

On enabling this option, the NetScaler will collect the time required to load and render the mainpage on the client.

Possible values: ENABLED, DISABLED

Default value: DISABLED

comment

Any comments about this action. In the CLI, if including spaces between words, enclose the comment in quotation marks. (The quotation marks are not required in the configuration utility.)

Example

```
add appflow action appflow_action_1 -collectors col1 col2
```

rm appflow action

Removes a configured AppFlow action. You cannot remove an action that is associated with an AppFlow policy.

Synopsis

```
rm appflow action <name>
```

Arguments

name

Name of the action to be removed.

Example

```
rm appflow action appflow_action_1
```

set appflow action

Modifies the specified parameters of an AppFlow action.

Synopsys

```
set appflow action <name> [-collectors <string> ...] [-clientSideMeasurements ( ENABLED | DISABLED )] [-comment <string>]
```

Arguments

name

Name of the action to be modified.

collectors

Name(s) of collector(s) to be associated with the AppFlow action.

clientSideMeasurements

On enabling this option, the NetScaler will collect the time required to load and render the mainpage on the client.

Possible values: ENABLED, DISABLED

Default value: DISABLED

comment

Any comments about this action. In the CLI, if including spaces between words, enclose the comment in quotation marks. (The quotation marks are not required in the configuration utility.)

Example

```
set appflow action appflow_action_1 -collectors col1 col2 col3
```

unset appflow action

Use this command to remove appflow action settings. Refer to the set appflow action command for meanings of the arguments.

Synopsys

```
unset appflow action <name> [-clientSideMeasurements] [-comment]
```

rename appflow action

Renames an AppFlow action.

Synopsys

```
rename appflow action <name>@ <newName>@
```

Arguments

name

Existing name of the action.

newName

New name for the AppFlow action. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at

(@), equals (=), and hyphen (-) characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my appflow action" or 'my appflow action').

Example

```
rename appflow action old_name new_name
```

show appflow action

Displays information about AppFlow action(s), or about the specified AppFlow action.

Synopsys

```
show appflow action [<name>]
```

Arguments

name

Name of the action about which to display information.

Outputs

stateflag

hits

The number of times the action has been taken.

collectors

Name(s) of collector(s) to be associated with the AppFlow action.

clientSideMeasurements

On enabling this option, the NetScaler will collect the time required to load and render the mainpage on the client.

referenceCount

The number of references to the action.

description

Description of the action

comment

Comments associated with the AppFlow action.

devno

count

Example

```
1. show appflow action 2. show appflow action appflow_action_1
```

appflow collector

The following operations can be performed on "appflow collector":

[add](#) | [rm](#) | [rename](#) | [show](#)

add appflow collector

Adds a new AppFlow collector. A collector receives the flow records generated by the NetScaler appliance. You can add only four AppFlow collectors to the NetScaler appliance.

Synopsis

```
add appflow collector <name> -IPAddress <ip_addr> [-port <port>] [-netProfile <string>]
```

Arguments

name

Name for the collector. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at

(@), equals (=), and hyphen (-) characters.

Only four collectors can be configured.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my appflow collector" or 'my appflow collector').

IPAddress

IPv4 address of the collector.

port

UDP port on which the collector listens.

Default value: 4739

netProfile

Netprofile to associate with the collector. The IP address defined in the profile is used as the source IP address for AppFlow traffic for this collector. If you do not set this parameter, the NetScaler IP (NSIP) address is used as the source IP address.

Example

```
add appflow collector collector1 -IPAddress 192.168.1.40 -port 2055
```

rm appflow collector

Removes an AppFlow collector. You cannot remove a collector if it is associated with an AppFlow action.

Synopsis

```
rm appflow collector <name>
```

Arguments

name

Name of the collector to remove.

Example

```
rm appflow collector collector1
```

rename appflow collector

Renames an AppFlow collector.

Synopsys

```
rename appflow collector <name>@ <newName>@
```

Arguments

name

Existing name of the collector.

newName

New name for the collector. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at(@), equals (=), and hyphen (-) characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my appflow coll" or 'my appflow coll').

Example

```
rename appflow collector old_name new_name
```

show appflow collector

Displays information about all configured AppFlow collectors, or about the specified collector.

Synopsys

```
show appflow collector [<name>]
```

Arguments

name

Name of the collector about which to display information.

Outputs

IPAddress

IPv4 address of the collector.

port

UDP port on which the collector listens.

netProfile

Netprofile to associate with the collector. The IP address defined in the profile is used as the source IP address for AppFlow traffic for this collector. If you do not set this parameter, the NetScaler IP (NSIP) address is used as the source IP address.

devno

count

stateflag

Example

```
show appflow collector collector1
```

appflow global

The following operations can be performed on "appflow global":

[bind](#) | [unbind](#) | [show](#)

bind appflow global

Binds the AppFlow policy to one of the two global lists of AppFlow policies. A policy becomes active only after it is bound.

Synopsys

```
bind appflow global <policyName> <priority> [<gotoPriorityExpression>] [-type <type>] [-invoke (<labelType> <labelName>)]
```

Arguments

policyName

Name of the AppFlow policy to be bound.

priority

Integer specifying the priority of the policy. The lower the number, the higher the priority. By default, policies in the list are evaluated in the order of their priority numbers.

Minimum value: 1

Maximum value: 2147483647

gotoPriorityExpression

Expression or other value specifying the priority of the next policy, within the policy list, to evaluate if the current policy evaluates to TRUE. Specify one of the following values:

- * NEXT - Evaluate the policy with the next higher numbered priority.
 - * END - Stop evaluation.
 - * USE_INVOCATION_RESULT - Applicable if this policy invokes another policy list. If the final goto in the invoked policy list has a value of END, the evaluation stops. If the final goto is anything other than END, the current policy list performs a NEXT.
 - * An expression that evaluates to a number.
- If you specify an expression, it's evaluation result determines the next policy to evaluate, as follows:
- * If the expression evaluates to a higher numbered priority, that policy is evaluated next.
 - * If the expression evaluates to the priority of the current policy, the policy with the next higher priority number is evaluated next.
 - * If the expression evaluates to a priority number that is numerically higher than the highest priority number, policy evaluation ends.
- An UNDEF event is triggered if:
- * The expression is invalid.
 - * The expression evaluates to a priority number that is numerically lower than the current policy's priority.
 - * The expression evaluates to a priority number that is between the current policy's priority number (say, 30) and the highest priority number (say, 100), but does not match any configured priority number (for example, the expression evaluates to the number 85). This example assumes that the priority number increments by 10 for every successive policy, and therefore a priority number of 85 does not exist in the policy label.

type

The bind point to which to bind the policy.

Possible values: REQ_OVERRIDE, REQ_DEFAULT, OVERRIDE, DEFAULT, OTHERTCP_REQ_OVERRIDE, OTHERTCP_REQ_DEFAULT, MSSQL_REQ_OVERRIDE, MSSQL_REQ_DEFAULT, MYSQL_REQ_OVERRIDE, MYSQL_REQ_DEFAULT, ICA_REQ_OVERRIDE, ICA_REQ_DEFAULT, ORACLE_REQ_OVERRIDE, ORACLE_REQ_DEFAULT

invoke

Invoke policies bound to a virtual server or a user-defined policy label. After the invoked policies are evaluated, the flow returns to the policy with the next priority.

labelType

Type of policy label to invoke. Specify vserver for a policy label associated with a virtual server, or policylabel for a user-defined policy label.

Possible values: vserver, policylabel

labelName

Name of the label to invoke if the current policy evaluates to TRUE.

Example

```
i) bind appflow global pol9 9 ii) bind appflow global pol9 9 120 iii) bind appflow glo
```

unbind appflow global

Unbinds entities from an AppFlow global bind point.

Synopsis

```
unbind appflow global (<policyName> [-type <type>] [-priority <positive_integer>])
```

Arguments

policyName

Name of the policy to be unbound.

type

Bind point from which to unbind the policy.

Possible values: REQ_OVERRIDE, REQ_DEFAULT, OVERRIDE, DEFAULT, OTHERTCP_REQ_OVERRIDE, OTHERTCP_REQ_DEFAULT, MSSQL_REQ_OVERRIDE, MSSQL_REQ_DEFAULT, MYSQL_REQ_OVERRIDE, MYSQL_REQ_DEFAULT, ICA_REQ_OVERRIDE, ICA_REQ_DEFAULT, ORACLE_REQ_OVERRIDE, ORACLE_REQ_DEFAULT

priority

Priority of the NOPOLICY to be unbound. Applicable only if a NOPOLICY has been bound to the bind point.

Minimum value: 1

Maximum value: 2147483647

Example

```
unbind appflow global pol9
```

show appflow global

Displays the AppFlow global bind points and the number of policies bound to each global bind point, or more detailed information about the specified bind point.

Synopsys

show appflow global [-type <type>]

Arguments

type

Global bind point for which to show detailed information about the policies bound to the bind point.

Possible values: REQ_OVERRIDE, REQ_DEFAULT, OVERRIDE, DEFAULT, OTHERTCP_REQ_OVERRIDE, OTHERTCP_REQ_DEFAULT, MSSQL_REQ_OVERRIDE, MSSQL_REQ_DEFAULT, MYSQL_REQ_OVERRIDE, MYSQL_REQ_DEFAULT, ICA_REQ_OVERRIDE, ICA_REQ_DEFAULT, ORACLE_REQ_OVERRIDE, ORACLE_REQ_DEFAULT

Outputs

stateflag

policyName

Name of the AppFlow policy.

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

invoke

Invoke policies bound to a virtual server or a user-defined policy label. After the invoked policies are evaluated, the flow returns to the policy with the next priority.

labelType

Type of policy label to invoke. Specify vserver for a policy label associated with a virtual server, or policylabel for a user-defined policy label.

labelName

Name of the label to invoke if the current policy evaluates to TRUE.

numpol

The number of policies bound to the bindpoint.

flowType

Flow type of the bound AppFlow policy.

flags

devno

count

Example

```
show appflow global
```

appflow param

The following operations can be performed on "appflow param":

[set](#) | [unset](#) | [show](#)

set appflow param

Configures AppFlow parameters.

Synopsys

```
set appflow param [-templateRefresh <secs>] [-appnameRefresh <secs>] [-flowRecordInterval <secs>] [-udpPmtu <positive_integer>] [-httpUrl ( ENABLED | DISABLED )] [-AAAUserName ( ENABLED | DISABLED )] [-httpCookie ( ENABLED | DISABLED )] [-httpReferer ( ENABLED | DISABLED )] [-httpMethod ( ENABLED | DISABLED )] [-httpHost ( ENABLED | DISABLED )] [-httpUserAgent ( ENABLED | DISABLED )] [-clientTrafficOnly ( YES | NO )] [-httpContentType ( ENABLED | DISABLED )] [-httpAuthorization ( ENABLED | DISABLED )] [-httpVia ( ENABLED | DISABLED )] [-httpXForwardedFor ( ENABLED | DISABLED )] [-httpLocation ( ENABLED | DISABLED )] [-httpSetCookie ( ENABLED | DISABLED )] [-httpSetCookie2 ( ENABLED | DISABLED )] [-connectionChaining ( ENABLED | DISABLED )] [-httpDomain ( ENABLED | DISABLED )] [-skipCacheRedirectionHttpTransaction ( ENABLED | DISABLED )]
```

Arguments

templateRefresh

Refresh interval, in seconds, at which to export the template data. Because data transmission is in UDP, the templates must be resent at regular intervals.

Default value: 600

Minimum value: 60

Maximum value: 3600

appnameRefresh

Interval, in seconds, at which to send Appnames to the configured collectors. Appname refers to the name of an entity (virtual server, service, or service group) in the NetScaler appliance.

Default value: 600

Minimum value: 60

Maximum value: 3600

flowRecordInterval

Interval, in seconds, at which to send flow records to the configured collectors.

Default value: 60

Minimum value: 60

Maximum value: 3600

udpPmtu

MTU, in bytes, for IPFIX UDP packets.

Default value: 1472

Minimum value: 128

Maximum value: 1472

httpUrl

Include the http URL that the NetScaler appliance received from the client.

Possible values: ENABLED, DISABLED

Default value: DISABLED

AAAUserName

Enable AppFlow AAA Username logging.

Possible values: ENABLED, DISABLED

Default value: DISABLED

httpCookie

Include the cookie that was in the HTTP request the appliance received from the client.

Possible values: ENABLED, DISABLED

Default value: DISABLED

httpReferer

Include the web page that was last visited by the client.

Possible values: ENABLED, DISABLED

Default value: DISABLED

httpMethod

Include the method that was specified in the HTTP request that the appliance received from the client.

Possible values: ENABLED, DISABLED

Default value: DISABLED

httpHost

Include the host identified in the HTTP request that the appliance received from the client.

Possible values: ENABLED, DISABLED

Default value: DISABLED

httpUserAgent

Include the client application through which the HTTP request was received by the NetScaler appliance.

Possible values: ENABLED, DISABLED

Default value: DISABLED

clientTrafficOnly

Generate AppFlow records for only the traffic from the client.

Possible values: YES, NO

Default value: NO

httpContentType

Include the HTTP Content-Type header sent from the server to the client to determine the type of the content sent.

Possible values: ENABLED, DISABLED

Default value: DISABLED

httpAuthorization

Include the HTTP Authorization header information.

Possible values: ENABLED, DISABLED

Default value: DISABLED

httpVia

Include the httpVia header which contains the IP address of proxy server through which the client accessed the server.

Possible values: ENABLED, DISABLED

Default value: DISABLED

httpXForwardedFor

Include the httpXForwardedFor header, which contains the original IP Address of the client using a proxy server to access the server.

Possible values: ENABLED, DISABLED

Default value: DISABLED

httpLocation

Include the HTTP location headers returned from the HTTP responses.

Possible values: ENABLED, DISABLED

Default value: DISABLED

httpSetCookie

Include the Set-cookie header sent from the server to the client in response to a HTTP request.

Possible values: ENABLED, DISABLED

Default value: DISABLED

httpSetCookie2

Include the Set-cookie header sent from the server to the client in response to a HTTP request.

Possible values: ENABLED, DISABLED

Default value: DISABLED

connectionChaining

Enable connection chaining so that the client server flows of a connection are linked. Also the connection chain ID is propagated across NetScalers, so that in a multi-hop environment the flows belonging to the same logical connection are linked. This id is also logged as part of appflow record

Possible values: ENABLED, DISABLED

Default value: DISABLED

httpDomain

Include the http domain request to be exported.

Possible values: ENABLED, DISABLED

Default value: DISABLED

skipCacheRedirectionHttpTransaction

Skip Cache http transaction. This HTTP transaction is specific to Cache Redirection module. In Case of Cache Miss there will be another HTTP transaction initiated by the cache server.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
set appflow param -templateRefresh 240
```

unset appflow param

Use this command to remove appflow param settings. Refer to the set appflow param command for meanings of the arguments.

Synopsys

```
unset appflow param [-templateRefresh] [-appnameRefresh] [-flowRecordInterval] [-udpPmtu] [-httpUrl] [-AAAUserName] [-httpCookie] [-httpReferer] [-httpMethod] [-httpHost] [-httpUserAgent] [-clientTrafficOnly] [-httpContentType] [-httpAuthorization] [-httpVia] [-httpXForwardedFor] [-httpLocation] [-httpSetCookie] [-httpSetCookie2] [-connectionChaining] [-httpDomain] [-skipCacheRedirectionHttpTransaction]
```

show appflow param

Displays AppFlow parameters.

Synopsys

```
show appflow param
```

Outputs

templateRefresh

Refresh interval, in seconds, at which to export the template data. Because data transmission is in UDP, the templates must be resent at regular intervals.

appnameRefresh

Interval, in seconds, at which to send Appnames to the configured collectors. Appname refers to the name of an entity (virtual server, service, or service group) in the NetScaler appliance.

flowRecordInterval

Interval, in seconds, at which to send flow records to the configured collectors.

udpPmtu

MTU, in bytes, for IPFIX UDP packets.

httpUrl

State of AppFlow HTTP URL logging.

AAAUserName

State of AppFlow AAA User logging.

httpCookie

State of AppFlow HTTP cookie logging.

httpReferer

State of AppFlow HTTP referer logging.

httpMethod

State of AppFlow HTTP method logging.

httpHost

State of AppFlow HTTP host logging.

httpUserAgent

State of AppFlow HTTP user-agent logging.

clientTrafficOnly

Generate AppFlow records for only the traffic from the client.

httpContentType

State of AppFlow HTTP Content-Type header logging

httpAuthorization

State of AppFlow HTTP Authorization header logging

httpVia

State of AppFlow HTTP Via header logging

httpXForwardedFor

State of AppFlow HTTP X-Forwarded-For header logging

httpLocation

State of AppFlow HTTP Location header logging

httpSetCookie

State of AppFlow HTTP Setcookie header logging

httpSetCookie2

State of AppFlow HTTP Setcookie2 header logging

connectionChaining

State of connection-chaining feature

httpDomain

State of AppFlow HTTP Domain name logging

skipCacheRedirectionHttpTransaction

Skip Cache http transaction. This HTTP transaction is specific to Cache Redirection module. In Case of Cache Miss there will be another HTTP transaction initiated by the cache server.

appflow policy

The following operations can be performed on "appflow policy":

add | **rm** | **set** | **unset** | **rename** | **show**

add appflow policy

Adds an Appflow policy. The policy specifies the rule based on which the traffic is evaluated, and the action to be taken if the rule returns "TRUE".

Synopsys

add appflow policy <name> <rule> <action> [-comment <string>]

Arguments

name

Name for the policy. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at

(@), equals (=), and hyphen (-) characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my appflow policy" or 'my appflow policy').

rule

Expression or other value against which the traffic is evaluated. Must be a Boolean, default syntax expression. Note:

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters> + <string of 245 characters>"

The following requirements apply only to the NetScaler CLI:

- * If the expression includes one or more spaces, enclose the entire expression in double quotation marks.
- * If the expression itself includes double quotation marks, escape the quotations by using the \ character.
- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Name of the action to be associated with this policy.

comment

Any comments about this policy.

Example

```
add appflow policy appflow_pol "HTTP.REQ.HEADER(\\\\"header\\\\" ).CONTAINS(\\\\"qh3\\\\" )"
```

rm appflow policy

Removes an AppFlow policy. (Cannot remove a policy that is bound to a policy label.)

Synopsys

rm appflow policy <name>

Arguments

name

Name of the policy to be removed.

Example

```
rm appflow policy appflow_policy_1
```

set appflow policy

Modifies the rule and/or action for an existing AppFlow policy. The rule for flow type can be changed only if the associated action is of NEUTRAL flow type.

Synopsys

```
set appflow policy <name> [-rule <expression>] [-action <string>] [-comment <string>]
```

Arguments

name

Name of the policy to modify.

rule

Expression or other value against which the traffic is evaluated. Must be a Boolean, default syntax expression. Note:

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

The following requirements apply only to the NetScaler CLI:

- * If the expression includes one or more spaces, enclose the entire expression in double quotation marks.
- * If the expression itself includes double quotation marks, escape the quotations by using the \\ character.
- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Name of the action to be associated with this policy.

comment

Any comments about this policy.

Example

```
set appflow policy appflow_policy -rule "HTTP.REQ.HEADER(\\\\"header\\\\" ).CONTAINS(\\\\""
```

unset appflow policy

Use this command to remove appflow policy settings. Refer to the set appflow policy command for meanings of the arguments.

Synopsys

```
unset appflow policy <name> -comment
```

rename appflow policy

Renames an AppFlow policy.

Synopsys

```
rename appflow policy <name>@ <newName>@
```

Arguments

name

Existing name of the policy.

newName

New name for the policy. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my appflow policy" or 'my appflow policy').

Example

```
rename appflow policy old_name new_name
```

show appflow policy

Displays information about all configured AppFlow policies, or detailed information about the specified policy.

Synopsys

```
show appflow policy [<name>]
```

Arguments

name

Name of the policy about which to display detailed information.

Outputs

stateflag

rule

Expression to be used by AppFlow policy.

action

AppFlow action associated with the policy.

hits

Number of hits.

undefHits

Number of policy UNDEF hits.

activePolicy

Indicates whether policy is bound or not.

boundTo

Location where policy is bound.

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

labelType

Type of policy label invocation.

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

description

Description of the policy

comment

Any comments about this policy.

bindPolicyType

vserverType

devno

count

Example

```
show appflow policy
```

appflow policylabel

The following operations can be performed on "appflow policylabel":

[add](#) | [rm](#) | [bind](#) | [unbind](#) | [rename](#) | [show](#)

add appflow policylabel

Creates a user-defined AppFlow policy label. You can bind AppFlow policies to the AppFlow policy label.

Synopsis

```
add appflow policylabel <labelName> [-policylabeltype ( HTTP | OTHERTCP )]
```

Arguments

labelName

Name of the AppFlow policy label. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at

(@), equals (=), and hyphen (-) characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my appflow policylabel" or 'my appflow policylabel').

policylabeltype

Type of traffic evaluated by the policies bound to the policy label.

Possible values: HTTP, OTHERTCP

Default value: HTTP

Example

```
add appflow policylabel appflow_pol_label
```

rm appflow policylabel

Removes an AppFlow policy label.

Synopsis

```
rm appflow policylabel <labelName>
```

Arguments

labelName

Name of the policy label to be removed.

Example

```
rm appflow policylabel appflow_pol_label
```

bind appflow policylabel

Binds an AppFlow policy to an AppFlow policy label.

Synopsys

bind appflow policylabel <labelName> -policyName <string> -priority <positive_integer> [-gotoPriorityExpression <expression>] [-invoke (<labelType> <labelName>)]

Arguments

labelName

Name of the label to invoke if the current policy evaluates to TRUE.

policyName

Name of the policy to bind to the policy label.

priority

Priority assigned to the policy. The lower the number, the higher the priority.

Minimum value: 1

Maximum value: 2147483647

gotoPriorityExpression

Expression or other value specifying the priority of the next policy, within the policy label, to evaluate if the current policy evaluates to TRUE. Specify one of the following values:

- * NEXT - Evaluate the policy with the next higher numbered priority.
- * END - Stop evaluation.
- * USE_INVOCATION_RESULT - Applicable if this policy invokes another policy label. If the final goto in the invoked policy label has a value of END, the evaluation stops. If the final goto is anything other than END, the current policy label performs a NEXT.
- * An expression that evaluates to a number.

If you specify an expression, it's evaluation result determines the next policy to evaluate, as follows:

- * If the expression evaluates to a higher numbered priority, that policy is evaluated next.
- * If the expression evaluates to the priority of the current policy, the policy with the next higher priority number is evaluated next.
- * If the expression evaluates to a priority number that is numerically higher than the highest priority number, policy evaluation ends.

An UNDEF event is triggered if:

- * The expression is invalid.
- * The expression evaluates to a priority number that is numerically lower than the current policy's priority.
- * The expression evaluates to a priority number that is between the current policy's priority number (say, 30) and the highest priority number (say, 100), but does not match any configured priority number (for example, the expression evaluates to the number 85). This example assumes that the priority number increments by 10 for every successive policy, and therefore a priority number of 85 does not exist in the policy label.

invoke

Invoke policies bound to a virtual server or a user-defined policy label. After the invoked policies are evaluated, the flow returns to the policy with the next priority.

labelType

Type of policy label to be invoked.

Possible values: vserver, policylabel

Example

```
bind appflow policylabel appflow_pol_label -policyName appflow_pol -priority 1
```

unbind appflow policylabel

Unbinds an AppFlow policy from an AppFlow policy label.

Synopsys

```
unbind appflow policylabel <labelName> <policyName> [-priority <positive_integer>]
```

Arguments

labelName

Name of the policy label from which to unbind a policy.

policyName

Name of the policy to unbind.

priority

Priority of the NOPOLICY to be unbound. Applicable only if a NOPOLICY has been bound to the policy label.

Minimum value: 1

Maximum value: 2147483647

Example

```
unbind appflow policylabel appflow_pol_label appflow_pol
```

rename appflow policylabel

Renames an AppFlow policy label.

Synopsys

```
rename appflow policylabel <labelName>@ <newName>@
```

Arguments

labelName

Existing name of the policylabel.

newName

New name for the policy label. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my appflow policylabel" or 'my appflow policylabel')

Example

```
rename appflow policylabel old_name new_name
```

show appflow policylabel

Displays information about all AppFlow policy labels, or detailed information about the specified policy label.

Synopsys

show appflow policylabel [<labelName>]

Arguments

labelName

Name of the policy label about which to display detailed information.

Outputs

stateflag

policylabeltype

Type of traffic evaluated by the policies bound to the policy label.

numpol

Number of policies bound to the policy label.

hits

Number of times the policy label was invoked.

policyName

Name of the AppFlow policy.

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

invoke

Invoke policies bound to a virtual server or a user-defined policy label. After the invoked policies are evaluated, the flow returns to the policy with the next priority.

labelType

Type of policy label to be invoked.

labelName

Name of the label to invoke if the current policy evaluates to TRUE.

flowType

Flowtype of the bound AppFlow policy.

description

Description of the policylabel

flags

devno

count

Example

i) `show appflow policylabel appflow_pol_label` ii) `show appflow policylabel`

Application Firewall Commands

The entities on which you can perform NetScaler CLI operations:

- o appfw
- o appfw JSONContentType
- o appfw XMLContentType
- o appfw archive
- o appfw confidField
- o appfw customSettings
- o appfw fieldType
- o appfw global
- o appfw htmlerrorpage
- o appfw learningdata
- o appfw learningsettings
- o appfw policy
- o appfw policylabel
- o appfw profile
- o appfw settings
- o appfw signatures
- o appfw stats
- o appfw transactionRecords
- o appfw wsdl
- o appfw xmlerrorpage
- o appfw xmlschema

appfw

The following operations can be performed on "appfw":

stat appfw

Displays application firewall statistics.

Synopsys

```
stat appfw [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

total violations (totviols)

Total number of security check violations seen by the Application Firewall.

Recent Ave Response Time (ms) (shortAvgRespTime)

Average backend response time in milliseconds over the last 7 seconds

Long Term Ave Response Time (ms) (longAvgRespTime)

Average backend response time in milliseconds since reboot

requests (reqs)

HTTP/HTTPS requests sent to your protected web servers via the Application Firewall.

Request Bytes (reqBytes)

Number of bytes transfered for requests

responses (resps)

HTTP/HTTPS responses sent by your protected web servers via the Application Firewall.

Response Bytes (resBytes)

Number of bytes transferred for responses

aborts

Incomplete HTTP/HTTPS requests aborted by the client before the Application Firewall could finish processing them.

redirects (redirect)

HTTP/HTTPS requests redirected by the Application Firewall to a different Web page or web server. (HTTP 302)

Traps Dropped (trapsDr)

AppFirewall SNMP traps dropped due to time limit.

start URL (startURL)

Number of Start URL security check violations seen by the Application Firewall.

deny URL (denyURL)

Number of Deny URL security check violations seen by the Application Firewall.

referer header (refererHdr)

Number of Referer Header security check violations seen by the Application Firewall.

buffer overflow (bufovfl)

Number of Buffer Overflow security check violations seen by the Application Firewall.

cookie consistency (cookie)

Number of Cookie Consistency security check violations seen by the Application Firewall.

CSRF form tag (csrf_tag)

Number of Cross Site Request Forgery form tag security check violations seen by the Application Firewall.

HTML Cross-site scripting (xss)

Number of HTML Cross-Site Scripting security check violations seen by the Application Firewall.

HTML SQL injection (sql)

Number of HTML SQL Injection security check violations seen by the Application Firewall.

field format (fieldfmt)

Number of Field Format security check violations seen by the Application Firewall.

field consistency (fieldcon)

Number of Field Consistency security check violations seen by the Application Firewall.

credit card (ccard)

Number of Credit Card security check violations seen by the Application Firewall.

safe object (safeobj)

Number of Safe Object security check violations seen by the Application Firewall.

Signature Violations (sigs)

Number of Signature violations seen by the Application Firewall.

XML Format (wfcViolations)

Number of XML Format security check violations seen by the Application Firewall.

XML Denial of Service (XDoS) (xdosViolations)

Number of XML Denial-of-Service security check violations seen by the Application Firewall.

XML Message Validation (msgvalViolations)

Number of XML Message Validation security check violations seen by the Application Firewall.

Web Services Interoperability (wsIViolations)

Number of Web Services Interoperability (WS-I) security check violations seen by the Application Firewall.

XML SQL Injection (xmlSqlViolations)

Number of XML SQL Injection security check violations seen by the Application Firewall.

XML Cross-Site Scripting (xmlXssViolations)

Number of XML Cross-Site Scripting (XSS) security check violations seen by the Application Firewall.

XML Attachment (xmlAttachmentViolations)

Number of XML Attachment security check violations seen by the Application Firewall.

SOAP Fault Violations (soapflt)

Number of requests returning soap:fault from the backend server

XML Generic Violations (genflt)

Number of requests returning XML generic error from the backend server

HTTP Client Errors (4xx Resp) (4xxResps)

Number of requests returning HTTP 4xx from the backend server

HTTP Server Errors (5xx Resp) (5xxResps)

Number of requests returning HTTP 5xx from the backend server

appfw JSONContentType

The following operations can be performed on "appfw JSONContentType":

[add](#) | [rm](#) | [show](#)

add appfw JSONContentType

Add JSON content type. This will classify a request/response with the specified content type as JSON

Synopsis

```
add appfw JSONContentType <JSONContenttypevalue> [-isRegex ( REGEX | NOTREGEX )]
```

Arguments

JSONContenttypevalue

Content type to be classified as JSON

isRegex

Is json content type a regular expression?

Possible values: REGEX, NOTREGEX

Default value: NOTREGEX

rm appfw JSONContentType

Remove JSON content type.

Synopsis

```
rm appfw JSONContentType <JSONContenttypevalue>
```

Arguments

JSONContenttypevalue

Content type to be classified as JSON

show appfw JSONContentType

Display all JSON content types.

Synopsis

```
show appfw JSONContentType [<JSONContenttypevalue>]
```

Arguments

JSONContenttypevalue

Content type to be classified as JSON

Outputs

isRegex

Is json content type a regular expression?

builtin

Flag to determine if jsoncontenttype is built-in or not

devno

count

stateflag

appfw XMLContentType

The following operations can be performed on "appfw XMLContentType":

[add](#) | [rm](#) | [show](#)

add appfw XMLContentType

Add XML content type. This will classify a request/response with the specified content type as XML

Synopsis

```
add appfw XMLContentType <XMLContenttypevalue> [-isRegex ( REGEX | NOTREGEX )]
```

Arguments

XMLContenttypevalue

Content type to be classified as XML

isRegex

Is field name a regular expression?

Possible values: REGEX, NOTREGEX

Default value: NOTREGEX

rm appfw XMLContentType

Remove XML content type.

Synopsis

```
rm appfw XMLContentType <XMLContenttypevalue>
```

Arguments

XMLContenttypevalue

Content type to be classified as XML

show appfw XMLContentType

Display all xml content types.

Synopsis

```
show appfw XMLContentType [<XMLContenttypevalue>]
```

Arguments

XMLContenttypevalue

Content type to be classified as XML

Outputs

isRegex

Is field name a regular expression?

builtin

Flag to determine if xmlcontenttype is built-in or not

devno

count

stateflag

appfw archive

The following operations can be performed on "appfw archive":

[show](#) | [export](#) | [import](#) | [rm](#)

show appfw archive

Synopsys

show appfw archive

Outputs

response

Example

```
show appfw archive
```

export appfw archive

Exports the archive file to the specified location

Synopsys

export appfw archive <name> <target>

Arguments

name

Name of tar archive

target

Path to the file to be exported

import appfw archive

Imports the archive file from specified location

Synopsys

import appfw archive <src> <name> [-comment <string>]

Arguments

src

Indicates the source of the tar archive file as a URL
of the form

<protocol>://<host>[:<port>][/<path>]

<protocol> is http or https.

<host> is the DNS name or IP address of the http or https server.

<port> is the port number of the server. If omitted, the

default port for http or https will be used.

<path> is the path of the file on the server.

Import will fail if an https server requires client certificate authentication.

name

Indicates name of archive

comment

Comments associated with this archive.

rm appfw archive

Removes the archive created by archive command.

Synopsys

rm appfw archive <name>

Arguments

name

Indicates name of the archive to be removed.

Example

```
rm appfw archive <name>
```

appfw confidField

The following operations can be performed on "appfw confidField":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add appfw confidField

Defines the specified web form field as confidential. Form fields designated as confidential have the information that is provided in those fields x'd out in the audit logs.

Synopsys

```
add appfw confidField <fieldName> <url> [-isRegex ( REGEX | NOTREGEX )] [-comment <string>] [-state ( ENABLED | DISABLED )]
```

Arguments

fieldName

Name of the form field to designate as confidential.

url

URL of the web page that contains the web form.

isRegex

Method of specifying the form field name. Available settings function as follows:

* REGEX. Form field is a regular expression.

* NOTREGEX. Form field is a literal string.

Possible values: REGEX, NOTREGEX

Default value: NOTREGEX

comment

Any comments to preserve information about the form field designation.

state

Enable or disable the confidential field designation.

Possible values: ENABLED, DISABLED

Default value: ENABLED

rm appfw confidField

Removes a confidential field designation.

Synopsys

```
rm appfw confidField <fieldName> <url>
```

Arguments

fieldName

Name of the web form field.

url

URL of the web page that contains the web form in which the field appears.

set appfw confidField

Modifies the specified parameters of a confidential field setting. Form fields designated as confidential have the information that is provided in those fields x'd out in the audit logs.

Synopsys

```
set appfw confidField <fieldName> <url> [-comment <string>] [-isRegex ( REGEX | NOTREGEX )] [-state ( ENABLED | DISABLED )]
```

Arguments

fieldName

Name of the field to modify.

url

URL of the web page that contains the web form.

comment

Any comments to preserve information about the form field designation.

isRegex

Method of specifying the form field name. Available settings function as follows:

* REGEX. Form field is a regular expression.

* NOTREGEX. Form field is a literal string.

Possible values: REGEX, NOTREGEX

Default value: NOTREGEX

state

Enable or disable the confidential field designation.

Possible values: ENABLED, DISABLED

Default value: ENABLED

unset appfw confidField

Use this command to remove appfw confidField settings. Refer to the set appfw confidField command for meanings of the arguments.

Synopsys

```
unset appfw confidField <fieldName> <url> [-comment] [-isRegex] [-state]
```

show appfw confidField

Displays the current settings for the specified application firewall confidential field designation. If no confidential field designation is specified, displays a list of all application firewall confidential field designations on the NetScaler appliance.

Synopsys

```
show appfw confidField [<fieldName> <url>]
```

Arguments

fieldName

Name of the web form field.

url

URL of the web page that contains the web form with the form field.

Outputs

isRegex

Method of specifying the form field name. Available settings function as follows:

- * REGEX. Form field is a regular expression.
- * NOTREGEX. Form field is a literal string.

comment

Any comments to preserve information about the form field designation.

state

Enable or disable the confidential field designation.

devno**count****stateflag**

appfw customSettings

The following operations can be performed on "appfw customSettings":

[export](#) | [rm](#) | [show](#) | [import](#) | [update](#)

export appfw customSettings

NOTE: This command is deprecated.Changed CLI commands for Appfw "customSettings" to "signatures"

Synopsys

Arguments

name

target

rm appfw customSettings

Removes the object imported by import customsettings. NOTE: This command is deprecated.Changed CLI commands for Appfw "customSettings" to "signatures"

Synopsys

Arguments

name

Indicates name of custom-settings object.

Example

```
rm customsettings <name>
```

show appfw customSettings

Displays the object imported by import customsettings. NOTE: This command is deprecated.Changed CLI commands for Appfw "customSettings" to "signatures"

Synopsys

Arguments

name

Outputs

response

Example

```
show appfw customsettings
```

import appfw customSettings

Downloads the Application Firewall Custom Settings XML configuration to the NetScaler Box with the given object name
NOTE: This command is deprecated.Changed CLI commands for Appfw "customSettings" to "signatures"

Synopsys

Arguments

src

Indicates the source of the custom settings file as a URL

of the form

<protocol>://<host>[:<port>][/<path>]

<protocol> is http or https.

<host> is the DNS name or IP address of the http or https server.

<port> is the port number of the server. If omitted, the

default port for http or https will be used.

<path> is the path of the file on the server.

Import will fail if an https server requires client

certificate authentication.

name

Indicates name of custom-settings object.

comment

Comments.

overwrite

Overwrites the existing file

xslt

XSLT file URL.

merge

Merges the existing Signature with new signature rules

sha1

File path for sha1 file to validate signature file

Example

```
import customsettings http://www.example.com/ns/customsettings.xml my-settings
```

update appfw customSettings

Updates the Application Firewall Custom Settings XML configuration to the NetScaler Box with the given object name
NOTE: This command is deprecated.Changed CLI commands for Appfw "customSettings" to "signatures"

Synopsys

Arguments

name

Indicates name of the custom-settings object to update.

mergeDefault

Merges signature file with default signature file.

Example

```
update customsettings my-settings
```

appfw fieldType

The following operations can be performed on "appfw fieldType":

[add](#) | [rm](#) | [set](#) | [show](#)

add appfw fieldType

Adds a field type to the list of field types used by the field format security check. A field type is a regular expression defining the type of data that can appear in a web form field. The Learning engine also uses the field types list to generate appropriate field type assignments for the field formats check.

Synopsis

add appfw fieldType <name> <regex> <priority> [-comment <string>]

Arguments

name

Name for the field type.

Must begin with a letter, number, or the underscore character `[_]`, and must contain only letters, numbers, and the hyphen `[-]`, period `[.]`, pound `[\#]`, space `[]`, at `[@]`, equals `[=]`, colon `[:]`, and underscore characters. Cannot be changed after the field type is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks `"my field type"` or `'my field type'`.

regex

PCRE - format regular expression defining the characters and length allowed for this field type.

priority

Positive integer specifying the priority of the field type. A lower number specified a higher priority. Field types are checked in the order of their priority numbers.

Minimum value: 0

Maximum value: 64000

comment

Comment describing the type of field that this field type is intended to match.

rm appfw fieldType

Removes an application firewall field type.

Synopsis

rm appfw fieldType <name>

Arguments

name

Name of the field type.

set appfw fieldType

Modifies the properties of the specified application firewall field type.

Synopsys

set appfw fieldType <name> <regex> <priority> [-comment <string>]

Arguments

name

Name for the field type.

Must begin with a letter, number, or the underscore character `\"_\"`, and must contain only letters, numbers, and the hyphen `\"-\"`, period `\".\"`, pound `\"#\"`, space `\" \"`, at `\"@\"`, equals `\"=\"`, colon `\":\"`, and underscore characters. Cannot be changed after the field type is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks `\"(for example, \"my field type\" or 'my field type')\"`.

regex

PCRE - format regular expression defining the characters and length allowed for this field type.

priority

Positive integer specifying the priority of the field type. A lower number specified a higher priority. Field types are checked in the order of their priority numbers.

Minimum value: 0

Maximum value: 64000

comment

Comment describing the type of field that this field type is intended to match.

show appfw fieldType

Displays the regular expression that defines the specified field type and its priority. If no field type is specified, displays all form field types currently configured on the NetScaler appliance.

Synopsys

show appfw fieldType [<name>]

Arguments

name

Name of the field type.

Outputs

regex

PCRE - format regular expression defining the characters and length allowed for this field type.

priority

Positive integer specifying the priority of the field type. A lower number specified a higher priority. Field types are checked in the order of their priority numbers.

comment

Comment describing the type of field that this field type is intended to match.

builtin

Flag to determine if fieldtype is built-in or not

devno

count

stateflag

appfw global

The following operations can be performed on "appfw global":

[bind](#) | [unbind](#) | [show](#)

bind appfw global

Activates an application firewall policy.

Synopsys

```
bind appfw global <policyName> <priority> [-state ( ENABLED | DISABLED )] [<gotoPriorityExpression>] [-type <type>] [-invoke (<labelType> <labelName>)]
```

Arguments

policyName

Name of the policy.

priority

Positive integer specifying the priority of the policy. A lower number specifies a higher priority. Must be unique within the group of policies that are bound to the global bind point. Policies are evaluated in the order of their priority numbers.

Minimum value: 0

Maximum value: 2147483647

state

Enable or disable the binding to activate or deactivate the policy. This is applicable to classic policies only.

Possible values: ENABLED, DISABLED

Default value: ENABLED

gotoPriorityExpression

Expression or other value specifying the next policy to evaluate if the current policy evaluates to TRUE. Specify one of the following values:

- * NEXT - Evaluate the policy with the next higher priority number.
- * END - End policy evaluation.
- * USE_INVOCATION_RESULT - Applicable if this policy invokes another policy label. If the final goto in the invoked policy label has a value of END, the evaluation stops. If the final goto is anything other than END, the current policy label performs a NEXT.
- * A default syntax or classic expression that evaluates to a number.

If you specify an expression, the number to which it evaluates determines the next policy to evaluate, as follows:

- * If the expression evaluates to a higher numbered priority, the policy with that priority is evaluated next.
- * If the expression evaluates to the priority of the current policy, the policy with the next higher numbered priority is evaluated next.
- * If the expression evaluates to a number that is larger than the largest numbered priority, policy evaluation ends.

An UNDEF event is triggered if:

- * The expression is invalid.

* The expression evaluates to a priority number that is smaller than the current policy's priority number.

* The expression evaluates to a priority number that is between the current policy's priority number (say, 30) and the highest priority number (say, 100), but does not match any configured priority number (for example, the expression evaluates to the number 85). This example assumes that the priority number increments by 10 for every successive policy, and therefore a priority number of 85 does not exist in the policy label.

type

Bind point to which to bind the policy. Can be used only with NetScaler default policies. NetScaler classic policies are not supported. Available settings function as follows:

* REQ_OVERRIDE. Request override. Binds the policy to the priority request queue.

* REQ_DEFAULT. Binds the policy to the default request queue.

Possible values: REQ_OVERRIDE, REQ_DEFAULT, NONE

invoke

If the current policy evaluates to TRUE, terminate evaluation of policies bound to the current policy label, and then forward the request to the specified virtual server or evaluate the specified policy label.

labelType

Type of policy label to invoke if the current policy evaluates to TRUE and the invoke parameter is set. Available settings function as follows:

* reqvserver. Invoke the unnamed policy label associated with the specified request virtual server.

* policylabel. Invoke the specified user-defined policy label.

Possible values: reqvserver, policylabel

labelName

Name of the policy label to invoke if the current policy evaluates to TRUE, the invoke parameter is set, and Label Type is set to Policy Label.

unbind appfw global

Deactivates the specified application firewall policy. See the bind appfw policy command for descriptions of the parameters.

Synopsys

```
unbind appfw global <policyName> [-type <type>] [-priority <positive_integer>]
```

Arguments

policyName

Application Firewall policy name.

type

The bindpoint from which the policy is to be unbound.

Possible values: REQ_OVERRIDE, REQ_DEFAULT, NONE

priority

Priority of the NOPOLICY to be unbound.

Minimum value: 1

Maximum value: 2147483647

show appfw global

Displays a list of application firewall policies that are bound to the specified bind point. If no bind point is specified, displays a list of all application firewall policies

Synopsys

show appfw global [-type <type>]

Arguments

type

Bind point to which to policy is bound.

Possible values: REQ_OVERRIDE, REQ_DEFAULT, NONE

Outputs

policyName

Name of the policy.

priority

The priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

invoke

If the current policy evaluates to TRUE, terminate evaluation of policies bound to the current policy label, and then forward the request to the specified virtual server or evaluate the specified policy label.

state

Enable or disable the binding to activate or deactivate the policy. This is applicable to classic policies only.

bindPolicyType

The type of the policy.

policySubType

stateflag

stateflag

labelType

Type of policy label invocation.

labelName

Name of the policy label to invoke if the current policy evaluates to TRUE, the invoke parameter is set, and Label Type is set to Policy Label.

flowType

flowtype of the bound application firewall policy.

numpol

The number of policies bound to the bindpoint.

flags

policyType

flag

devno

count

appfw htmlerrorpage

The following operations can be performed on "appfw htmlerrorpage":

[rm](#) | [show](#) | [import](#) | [update](#)

rm appfw htmlerrorpage

Removes the specified XML error object.

Synopsis

```
rm appfw htmlerrorpage <name>
```

Arguments

name

Name of the XML error object to remove.

Example

```
rm htmlerrorpage <name>
```

show appfw htmlerrorpage

Displays the specified HTML error object. If no HTML error object is specified, lists all HTML error objects on the NetScaler appliance.

Synopsis

```
show appfw htmlerrorpage [<name>]
```

Arguments

name

Name of the HTML error object.

Outputs

response

Example

```
show appfw htmlerrorpage
```

import appfw htmlerrorpage

Imports the specified HTML error page to the NetScaler appliance and assigns it the specified name.

Synopsis

```
import appfw htmlerrorpage <src> <name> [-comment <string>] [-overwrite]
```

Arguments

src

URL (protocol, host, path, and name) for the location at which to store the imported HTML error object.

NOTE: The import fails if the object to be imported is on an HTTPS server that requires client certificate authentication for access.

name

Name to assign to the HTML error object on the NetScaler appliance.

comment

Any comments to preserve information about the HTML error object.

overwrite

Overwrite any existing HTML error object of the same name.

Example

```
import htmlerrorpage http://www.example.com/errorpage.html my-html-error-page
```

update appfw htmlerrorpage

Updates the specified HTML error object from the source.

Synopsys

```
update appfw htmlerrorpage <name>
```

Arguments

name

Name of the HTML error page object to update.

Example

```
update htmlerrorpage my-html-error-page
```

appfw learningdata

The following operations can be performed on "appfw learningdata":

[rm](#) | [show](#) | [reset](#) | [export](#)

rm appfw learningdata

Removes unreviewed application firewall learning data for the specified application firewall profile.

Synopsys

```
rm appfw learningdata <profileName> (-startURL <expression> | -cookieConsistency <string> | (-fieldConsistency <string> <formActionURL>) | (-crossSiteScripting <string> <formActionURL> [<location>]) | (-SQLInjection <string> <formActionURL> [<location>]) | (-fieldFormat <string> <formActionURL>) | (-CSRFTag <expression> <CSRFFormOriginURL>) | -XMLDoSCheck <expression> | -XMLWSICheck <expression> | -XMLAttachmentCheck <expression>) [-TotalXMLRequests]
```

Arguments

profileName

Name of the profile.

startURL

Start URL configuration.

cookieConsistency

Cookie Name.

fieldConsistency

Form field name.

formActionURL

Form action URL.

crossSiteScripting

Cross-site scripting.

location

Location of sql injection exception - form field, header or cookie.

Possible values: FORMFIELD, HEADER, COOKIE

SQLInjection

Form field name.

fieldFormat

Field format name.

CSRFTag

CSRF Form Action URL

CSRFFormOriginURL

CSRF Form Origin URL.

XMLDoSCheck

XML Denial of Service check, one of

MaxAttributes

MaxAttributeNameLength

MaxAttributeValueLength

MaxElementNameLength

MaxFileSize

MinFileSize

MaxCDATALength

MaxElements

MaxElementDepth

MaxElementChildren

NumDTDs

NumProcessingInstructions

NumExternalEntities

MaxEntityExpansions

MaxEntityExpansionDepth

MaxNamespaces

MaxNamespaceUriLength

MaxSOAPArraySize

MaxSOAPArrayRank

XMLWSICheck

Web Services Interoperability Rule ID.

XMLAttachmentCheck

XML Attachment Content-Type.

TotalXMLRequests

Total XML requests.

show appfw learningdata

Displays the unreviewed application firewall learning data for the specified profile and security check.

Synopsys

show appfw learningdata <profileName> <securityCheck>

Arguments

profileName

Name of the profile.

securityCheck

Name of the security check.

Possible values: startURL, cookieConsistency, fieldConsistency, crossSiteScripting, SQLInjection, fieldFormat, CSRFtag, XMLDoSCheck, XMLWSICheck, XMLAttachmentCheck, TotalXMLRequests

Outputs

url

Learnt url

name

Learnt field name

fieldType

Learnt field type

fieldFormatMinLength

The minimum allowed length for data in this form field.

fieldFormatMaxLength

The maximum allowed length for data in this form field.

hits

Learnt entity hit count

data

Learned data.

devno

count

stateflag

reset appfw learningdata

Remove all databases. Make transaction count zero

Synopsys

reset appfw learningdata

export appfw learningdata

Export appfw learnt data in csv format to the location /var/learnt_data/

Synopsys

export appfw learningdata <profileName> <securityCheck> [-target <string>]

Arguments

profileName

Name of the profile.

securityCheck

Name of the security check.

Possible values: startURL, cookieConsistency, fieldConsistency, crossSiteScripting, SQLInjection, fieldFormat, CSRFtag, XMLDoSCheck, XMLWSICheck, XMLAttachmentCheck, TotalXMLRequests

target

Target filename for data to be exported.

appfw learningsettings

The following operations can be performed on "appfw learningsettings":

[set](#) | [unset](#) | [show](#)

set appfw learningsettings

Configures the application firewall learning settings for the specified profile.

Synopsys

```
set appfw learningsettings <profileName> [-startURLMinThreshold <positive_integer>] [-startURLPercentThreshold <positive_integer>] [-cookieConsistencyMinThreshold <positive_integer>] [-cookieConsistencyPercentThreshold <positive_integer>] [-CSRFtagMinThreshold <positive_integer>] [-CSRFtagPercentThreshold <positive_integer>] [-fieldConsistencyMinThreshold <positive_integer>] [-fieldConsistencyPercentThreshold <positive_integer>] [-crossSiteScriptingMinThreshold <positive_integer>] [-crossSiteScriptingPercentThreshold <positive_integer>] [-SQLInjectionMinThreshold <positive_integer>] [-SQLInjectionPercentThreshold <positive_integer>] [-fieldFormatMinThreshold <positive_integer>] [-fieldFormatPercentThreshold <positive_integer>] [-XMLWSIMinThreshold <positive_integer>] [-XMLWSIPercentThreshold <positive_integer>] [-XMLAttachmentMinThreshold <positive_integer>] [-XMLAttachmentPercentThreshold <positive_integer>]
```

Arguments

profileName

Name of the profile.

startURLMinThreshold

Minimum number of application firewall sessions that the learning engine must observe to learn start URLs.

Default value: 1

Minimum value: 1

startURLPercentThreshold

Minimum percentage of application firewall sessions that must contain a particular start URL pattern for the learning engine to learn that start URL.

Default value: 0

Minimum value: 0

Maximum value: 100

cookieConsistencyMinThreshold

Minimum number of application firewall sessions that the learning engine must observe to learn cookies.

Default value: 1

Minimum value: 1

cookieConsistencyPercentThreshold

Minimum percentage of application firewall sessions that must contain a particular cookie pattern for the learning engine to learn that cookie.

Default value: 0

Minimum value: 0

Maximum value: 100

CSRFtagMinThreshold

Minimum number of application firewall sessions that the learning engine must observe to learn cross-site request forgery (CSRF) tags.

Default value: 1

Minimum value: 1

CSRFtagPercentThreshold

Minimum percentage of application firewall sessions that must contain a particular CSRF tag for the learning engine to learn that CSRF tag.

Default value: 0

Minimum value: 0

Maximum value: 100

fieldConsistencyMinThreshold

Minimum number of application firewall sessions that the learning engine must observe to learn field consistency information.

Default value: 1

Minimum value: 1

fieldConsistencyPercentThreshold

Minimum percentage of application firewall sessions that must contain a particular field consistency pattern for the learning engine to learn that field consistency pattern.

Default value: 0

Minimum value: 0

Maximum value: 100

crossSiteScriptingMinThreshold

Minimum number of application firewall sessions that the learning engine must observe to learn HTML cross-site scripting patterns.

Default value: 1

Minimum value: 1

crossSiteScriptingPercentThreshold

Minimum percentage of application firewall sessions that must contain a particular cross-site scripting pattern for the learning engine to learn that cross-site scripting pattern.

Default value: 0

Minimum value: 0

Maximum value: 100

SQLInjectionMinThreshold

Minimum number of application firewall sessions that the learning engine must observe to learn HTML SQL injection patterns.

Default value: 1

Minimum value: 1

SQLInjectionPercentThreshold

Minimum percentage of application firewall sessions that must contain a particular HTML SQL injection pattern for the learning engine to learn that HTML SQL injection pattern.

Default value: 0

Minimum value: 0

Maximum value: 100

fieldFormatMinThreshold

Minimum number of application firewall sessions that the learning engine must observe to learn field formats.

Default value: 1

Minimum value: 1

fieldFormatPercentThreshold

Minimum percentage of application firewall sessions that must contain a particular web form field pattern for the learning engine to recommend a field format for that form field.

Default value: 0

Minimum value: 0

Maximum value: 100

XMLWSIMinThreshold

Minimum number of application firewall sessions that the learning engine must observe to learn web services interoperability (WSI) information.

Default value: 1

Minimum value: 1

XMLWSIPercentThreshold

Minimum percentage of application firewall sessions that must contain a particular pattern for the learning engine to learn a web services interoperability (WSI) pattern.

Default value: 0

Minimum value: 0

Maximum value: 100

XMLAttachmentMinThreshold

Minimum number of application firewall sessions that the learning engine must observe to learn XML attachment patterns.

Default value: 1

Minimum value: 1

XMLAttachmentPercentThreshold

Minimum percentage of application firewall sessions that must contain a particular XML attachment pattern for the learning engine to learn that XML attachment pattern.

Default value: 0

Minimum value: 0

Maximum value: 100

unset appfw learningsettings

Use this command to remove appfw learningsettings settings. Refer to the set appfw learningsettings command for meanings of the arguments.

Synopsys

unset appfw learningsettings <profileName> [-startURLMinThreshold] [-startURLPercentThreshold] [-cookieConsistencyMinThreshold] [-cookieConsistencyPercentThreshold] [-CSRFtagMinThreshold] [-CSRFtagPercentThreshold] [-fieldConsistencyMinThreshold] [-fieldConsistencyPercentThreshold] [-crossSiteScriptingMinThreshold] [-crossSiteScriptingPercentThreshold] [-SQLInjectionMinThreshold] [-SQLInjectionPercentThreshold] [-fieldFormatMinThreshold] [-fieldFormatPercentThreshold] [-XMLWSIMinThreshold] [-XMLWSIPercentThreshold] [-XMLAttachmentMinThreshold] [-XMLAttachmentPercentThreshold]

show appfw learningsettings

Displays the current application firewall learning settings for the specified profile. If no profile is specified, displays the current application firewall settings for all profiles on the NetScaler appliance.

Synopsys

show appfw learningsettings [<profileName>]

Arguments

profileName

Name of the profile.

Outputs

startURLMinThreshold

Minimum number of application firewall sessions that the learning engine must observe to learn start URLs.

startURLPercentThreshold

Minimum percentage of application firewall sessions that must contain a particular start URL pattern for the learning engine to learn that start URL.

cookieConsistencyMinThreshold

Minimum number of application firewall sessions that the learning engine must observe to learn cookies.

cookieConsistencyPercentThreshold

Minimum percentage of application firewall sessions that must contain a particular cookie pattern for the learning engine to learn that cookie.

CSRFtagMinThreshold

Minimum number of application firewall sessions that the learning engine must observe to learn cross-site request forgery (CSRF) tags.

CSRFtagPercentThreshold

Minimum percentage of application firewall sessions that must contain a particular CSRF tag for the learning engine to learn that CSRF tag.

fieldConsistencyMinThreshold

Minimum number of application firewall sessions that the learning engine must observe to learn field consistency information.

fieldConsistencyPercentThreshold

Minimum percentage of application firewall sessions that must contain a particular field consistency pattern for the learning engine to learn that field consistency pattern.

crossSiteScriptingMinThreshold

Minimum number of application firewall sessions that the learning engine must observe to learn HTML cross-site scripting patterns.

crossSiteScriptingPercentThreshold

Minimum percentage of application firewall sessions that must contain a particular cross-site scripting pattern for the learning engine to learn that cross-site scripting pattern.

SQLInjectionMinThreshold

Minimum number of application firewall sessions that the learning engine must observe to learn HTML SQL injection patterns.

SQLInjectionPercentThreshold

Minimum percentage of application firewall sessions that must contain a particular HTML SQL injection pattern for the learning engine to learn that HTML SQL injection pattern.

fieldFormatMinThreshold

Minimum number of application firewall sessions that the learning engine must observe to learn field formats.

fieldFormatPercentThreshold

Minimum percentage of application firewall sessions that must contain a particular web form field pattern for the learning engine to recommend a field format for that form field.

XMLWSIMinThreshold

Minimum threshold to learn XML Web Services Interoperability.

XMLWSIPercentThreshold

Minimum threshold (in percent) to learn XML Web Services Interoperability.

XMLAttachmentMinThreshold

Minimum threshold to learn XML Attachments.

XMLAttachmentPercentThreshold

Minimum threshold (in percent) to learn XML Attachments.

devno

count

stateflag

appfw policy

The following operations can be performed on "appfw policy":

add | **rm** | **set** | **unset** | **show** | **stat** | **rename**

add appfw policy

Creates an application firewall policy.

Synopsys

add appfw policy <name> <rule> <profileName> [-comment <string>] [-logAction <string>]

Arguments

name

Name for the policy.

Must begin with a letter, number, or the underscore character \(_\), and must contain only letters, numbers, and the hyphen \(-\), period \(. \), pound \(\#\), space \(\), at (@), equals \(\= \), colon \(: \), and underscore characters. Can be changed after the policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks \(" for example, "my policy" or 'my policy' \).

rule

Name of the NetScaler named rule, or a NetScaler default syntax expression, that the policy uses to determine whether to filter the connection through the application firewall with the designated profile.

profileName

Name of the application firewall profile to use if the policy matches.

comment

Any comments to preserve information about the policy for later reference.

logAction

Where to log information for connections that match this policy.

rm appfw policy

Removes an application firewall policy.

Synopsys

rm appfw policy <name>

Arguments

name

Name of the policy to remove.

set appfw policy

Modifies the specified parameters of an application firewall policy.

Synopsys

set appfw policy <name> [-rule <expression>] [-profileName <string>] [-comment <string>] [-logAction <string>]

Arguments

name

Name of the policy to modify.

rule

Name of the NetScaler named rule, or a NetScaler default syntax expression, that the policy uses to determine whether to filter the connection through the application firewall with the designated profile.

profileName

Name of the application firewall profile to use if the policy matches.

comment

Any comments to preserve information about the policy for later reference.

logAction

Where to log information for connections that match this policy.

Example

```
set transform policy pol9 -rule "HTTP.REQ.HEADER(\\\\"header\\\\" ).CONTAINS(\\\\"qh2\\\\" "
```

unset appfw policy

Removes the settings of an existing application firewall policy. Attributes for which a default value is available revert to their default values. See the set appfw policy command for a description of the parameters..Refer to the set appfw policy command for meanings of the arguments.

Synopsys

unset appfw policy <name> [-comment] [-logAction]

Example

```
unset transform policy pol9 -undefAction
```

show appfw policy

Displays the current settings for the specified application firewall policy. If no policy name is provided, displays a list of all application firewall policies currently configured on the NetScaler appliance.

Synopsys

show appfw policy [<name>]

Arguments

name

Name of the policy.

Outputs

stateflag

rule

Name of the NetScaler named rule, or a NetScaler default syntax expression, that the policy uses to determine whether to filter the connection through the application firewall with the designated profile.

profileName

Name of the application firewall profile to use if the policy matches.

hits

Number of hits.

piHits

Number of hits.

undefHits

Number of Undef hits.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

labelType

Type of policy label invocation.

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

comment

Any comments to preserve information about the policy for later reference.

logAction

Where to log information for connections that match this policy.

boundTo

The entity name to which policy is bound

activePolicy

Indicates whether policy is bound or not.

priority

Specifies the priority of the policy.

bindPolicyType**policyType****vserverType****devno****count**

stat appfw policy

Displays statistics for the specified application firewall policy. If no application firewall policy is specified, displays abbreviated statistics for all application firewall policies.

Synopsys

stat appfw policy [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

name

Name of the application firewall policy.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Policy hits (Hits)

Number of hits on the policy

Policy undef hits (Undefhits)

Number of undef hits on the policy

Example

```
stat appfw policy
```

rename appfw policy

Renames an application firewall policy.

Synopsys

```
rename appfw policy <name>@ <newName>@
```

Arguments

name

Existing name of the application firewall policy.

newName

New name for the policy. Must begin with a letter, number, or the underscore character (`_`), and must contain only letters, numbers, and the hyphen (`-`), period (`.`) pound (`#`), space (), at (`@`), equals (`=`), colon (`:`), and underscore characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my policy" or 'my policy').

Example

```
rename appfw policy oldname newname
```

appfw policylabel

The following operations can be performed on "appfw policylabel":

`add` | `rm` | `bind` | `unbind` | `show` | `stat` | `rename`

add appfw policylabel

Creates a user-defined application firewall policy label.

Synopsis

```
add appfw policylabel <labelName> <policylabeltype>
```

Arguments

labelName

Name for the policy label. Must begin with a letter, number, or the underscore character (`_`), and must contain only letters, numbers, and the hyphen (`-`), period (`.`) pound (`#`), space (), at (`@`), equals (`=`), colon (`:`), and underscore characters. Can be changed after the policy label is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my policy label" or 'my policy label').

policylabeltype

Type of transformations allowed by the policies bound to the label. Always `http_req` for application firewall policy labels.

Possible values: `http_req`

Example

```
add appfw policylabel appfw_label http_req
```

rm appfw policylabel

Removes the specified application firewall policy label.

Synopsis

```
rm appfw policylabel <labelName>
```

Arguments

labelName

Name of the application firewall policy label to remove.

Example

```
rm appfw policylabel appfw_label
```

bind appfw policylabel

Binds the specified application firewall policy to the specified policy label.

Synopsis

```
bind appfw policylabel <labelName> <policyName> <priority> [<gotoPriorityExpression>] [-invoke (<labelType>
<labelName>)]
```

Arguments

labelName

Name of the policy label to invoke if the current policy evaluates to TRUE, the invoke parameter is set, and Label Type is set to Policy Label.

policyName

Name of the application firewall policy to bind to the policy label.

priority

Priority with which the policy is to be bound.

Minimum value: 1

Maximum value: 2147483647

gotoPriorityExpression

Expression or other value specifying the next policy to evaluate if the current policy evaluates to TRUE. Specify one of the following values:

- * NEXT - Evaluate the policy with the next higher priority number.
- * END - End policy evaluation.
- * USE_INVOCATION_RESULT - Applicable if this policy invokes another policy label. If the final goto in the invoked policy label has a value of END, the evaluation stops. If the final goto is anything other than END, the current policy label performs a NEXT.
- * A default syntax or classic expression that evaluates to a number.

If you specify an expression, the number to which it evaluates determines the next policy to evaluate, as follows:

- * If the expression evaluates to a higher numbered priority, the policy with that priority is evaluated next.
- * If the expression evaluates to the priority of the current policy, the policy with the next higher numbered priority is evaluated next.
- * If the expression evaluates to a number that is larger than the largest numbered priority, policy evaluation ends.

An UNDEF event is triggered if:

- * The expression is invalid.
- * The expression evaluates to a priority number that is smaller than the current policy's priority number.
- * The expression evaluates to a priority number that is between the current policy's priority number (say, 30) and the highest priority number (say, 100), but does not match any configured priority number (for example, the expression evaluates to the number 85). This example assumes that the priority number increments by 10 for every successive policy, and therefore a priority number of 85 does not exist in the policy label.

invoke

If the current policy evaluates to TRUE, terminate evaluation of policies bound to the current policy label, and then forward the request to the specified virtual server or evaluate the specified policy label.

labelType

Type of policy label to invoke if the current policy evaluates to TRUE and the invoke parameter is set. Available settings function as follows:

- * reqvserver. Invoke the unnamed policy label associated with the specified request virtual server.

* policylabel. Invoke the specified user-defined policy label.

Possible values: reqvserver, policylabel

Example

```
i) bind appfw policylabel trans_http_url pol_1 1 2 -invoke reqvserver CURRENT ii) bind :
```

unbind appfw policylabel

Unbinds the specified application firewall policy from the specified policy label. See the bind appfw policylabel command for descriptions of the parameters.

Synopsys

```
unbind appfw policylabel <labelName> <policyName> [-priority <positive_integer>]
```

Arguments

labelName

Name for the policy label. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.), pound (#), space (), at (@), equals (=), colon (:), and underscore characters. Can be changed after the policy label is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my policy label" or 'my policy label').

policyName

Name of the application firewall policy to bind to the policy label.

priority

Priority of the NOPOLICY to be unbound.

Minimum value: 1

Maximum value: 2147483647

Example

```
unbind appfw policylabel appfw_label
```

show appfw policylabel

Displays the current settings for the specified application firewall policy label. If no policy label is specified, displays a list of all application firewall policy labels currently configured on the NetScaler appliance.

Synopsys

```
show appfw policylabel [<labelName>]
```

Arguments

labelName

Name of the application firewall policy label.

Outputs

stateflag

policylabeltype

Type of transformations allowed by the policies bound to the label. Always http_req for application firewall policy labels.

numpol

Number of policies bound to label.

hits

Number of times policy label was invoked.

policyName

Name of the application firewall policy to bind to the policy label.

priority

Positive integer specifying the priority of the policy. A lower number specifies a higher priority. Must be unique within a group of policies that are bound to the same bind point or label. Policies are evaluated in the order of their priority numbers.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

invoke

If the current policy evaluates to TRUE, terminate evaluation of policies bound to the current policy label, and then forward the request to the specified virtual server or evaluate the specified policy label.

labelType

Type of policy label to invoke if the current policy evaluates to TRUE and the invoke parameter is set. Available settings function as follows:

- * reqvserver. Invoke the unnamed policy label associated with the specified request virtual server.
- * policylabel. Invoke the specified user-defined policy label.

labelName

Name of the policy label to invoke if the current policy evaluates to TRUE, the invoke parameter is set, and Label Type is set to Policy Label.

description

Description of the policylabel

flags**policyType****devno****count**

Example

```
i) show appfw policylabel appfw_label ii) show appfw policylabel
```

stat appfw policylabel

Displays statistics for the specified application firewall policy label. If no application firewall policy label is specified, displays abbreviated statistics for all application firewall policy labels.

Synopsys

stat appfw policylabel [<labelName>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

labelName

Name of the application firewall policy label.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Policy Label Hits (Hits)

Number of times policy label was invoked.

rename appfw policylabel

Renames an application firewall policy label.

Synopsys

rename appfw policylabel <labelName>@ <newName>@

Arguments

labelName

Existing name of the application firewall policy label.

newName

The new name of the application firewall policylabel.

Example

```
rename appfw policylabel oldname newname
```


appfw profile

The following operations can be performed on "appfw profile":

add | **rm** | **set** | **unset** | **bind** | **unbind** | **show** | **stat** | **archive** | **restore**

add appfw profile

Creates an application firewall profile, which specifies how the application firewall should protect a given type of web content. (A profile is equivalent to an action in other NetScaler features.)

Synopsys

```
add appfw profile <name> [-defaults ( basic | advanced )] [-startURLAction <startURLAction> ...] [-
contentTypeAction <contentTypeAction> ...] [-startURLClosure ( ON | OFF )] [-denyURLAction <denyURLAction> ...]
[-RefererHeaderCheck <RefererHeaderCheck>] [-cookieConsistencyAction <cookieConsistencyAction> ...] [-
cookieTransforms ( ON | OFF )] [-cookieEncryption <cookieEncryption>] [-cookieProxying ( none | sessionOnly )] [-
addCookieFlags <addCookieFlags>] [-fieldConsistencyAction <fieldConsistencyAction> ...] [-CSRFtagAction
<CSRFtagAction> ...] [-crossSiteScriptingAction <crossSiteScriptingAction> ...] [-
crossSiteScriptingTransformUnsafeHTML ( ON | OFF )] [-crossSiteScriptingCheckCompleteURLs ( ON | OFF )] [-
SQLInjectionAction <SQLInjectionAction> ...] [-SQLInjectionTransformSpecialChars ( ON | OFF )] [-
SQLInjectionType <SQLInjectionType>] [-SQLInjectionCheckSQLWildChars ( ON | OFF )] [-fieldFormatAction
<fieldFormatAction> ...] [-defaultFieldFormatType <string>] [-defaultFieldFormatMinLength <positive_integer>] [-
defaultFieldFormatMaxLength <positive_integer>] [-bufferOverflowAction <bufferOverflowAction> ...] [-
bufferOverflowMaxURLLength <positive_integer>] [-bufferOverflowMaxHeaderLength <positive_integer>] [-
bufferOverflowMaxCookieLength <positive_integer>] [-creditCardAction <creditCardAction> ...] [-creditCard
<creditCard> ...] [-creditCardMaxAllowed <positive_integer>] [-creditCardXOut ( ON | OFF )] [-requestContentType
<string>] [-responseContentType <string>] [-XMLDoSAction <XMLDoSAction> ...] [-XMLFormatAction
<XMLFormatAction> ...] [-XMLSQLInjectionAction <XMLSQLInjectionAction> ...] [-XMLSQLInjectionType
<XMLSQLInjectionType>] [-XMLSQLInjectionCheckSQLWildChars ( ON | OFF )] [-
XMLSQLInjectionParseComments <XMLSQLInjectionParseComments>] [-XMLXSSAction <XMLXSSAction> ...] [-
XMLWSIAction <XMLWSIAction> ...] [-XMLAttachmentAction <XMLAttachmentAction> ...] [-XMLValidationAction
<XMLValidationAction> ...] [-XMLErrorObject <string>] [-signatures <string>] [-XMLSOAPFaultAction
<XMLSOAPFaultAction> ...] [-useHTMLErrorObject ( ON | OFF )] [-errorURL <expression>] [-HTMLErrorObject
<string>] [-logEveryPolicyHit ( ON | OFF )] [-stripHtmlComments <stripHtmlComments>] [-stripXmlComments ( none
| all )] [-exemptClosureURLsFromSecurityChecks ( ON | OFF )] [-defaultCharSet <string>] [-postBodyLimit
<positive_integer>] [-fileUploadMaxNum <positive_integer>] [-canonicalizeHTMLResponse ( ON | OFF )] [-
enableFormTagging ( ON | OFF )] [-sessionlessFieldConsistency <sessionlessFieldConsistency>] [-
sessionlessURLClosure ( ON | OFF )] [-semicolonFieldSeparator ( ON | OFF )] [-excludeFileUploadFromChecks (
ON | OFF )] [-SQLInjectionParseComments <SQLInjectionParseComments>] [-invalidPercentHandling
<invalidPercentHandling>] [-type ( HTML | XML ) ...] [-checkRequestHeaders ( ON | OFF )] [-optimizePartialReqs (
ON | OFF )] [-URLDecodeRequestCookies ( ON | OFF )] [-comment <string>]
```

Arguments

name

Name for the profile. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.), pound (#), space (), at (@), equals (=), colon (:), and underscore (_) characters. Cannot be changed after the profile is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my profile" or 'my profile').

defaults

Default configuration to apply to the profile. Basic defaults are intended for standard content that requires little further configuration, such as static web site content. Advanced defaults are intended for specialized content that requires significant specialized configuration, such as heavily scripted or dynamic content.

CLI users: When adding an application firewall profile, you can set either the defaults or the type, but not both. To set both options, create the profile by using the add appfw profile command, and then use the set appfw profile command to configure the other option.

Possible values: basic, advanced

startURLAction

One or more Start URL actions. Available settings function as follows:

- * Block - Block connections that violate this security check.
- * Learn - Use the learning engine to generate a list of exceptions to this security check.
- * Log - Log violations of this security check.
- * Stats - Generate statistics for this security check.
- * None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -startURLaction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -startURLaction none".

Default value: AS_DEFAULT_DISPOSITION

contentTypeAction

One or more Content-type actions. Available settings function as follows:

- * Block - Block connections that violate this security check.
- * Log - Log violations of this security check.
- * None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -contentTypeaction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -contentTypeaction none".

Default value: AS_DEFAULT_CONTENT_TYPE_DISPOSITION

startURLClosure

Toggle the state of Start URL Closure.

Possible values: ON, OFF

Default value: OFF

denyURLAction

One or more Deny URL actions. Available settings function as follows:

- * Block - Block connections that violate this security check.
- * Log - Log violations of this security check.
- * Stats - Generate statistics for this security check.
- * None - Disable all actions for this security check.

NOTE: The Deny URL check takes precedence over the Start URL check. If you enable blocking for the Deny URL check, the application firewall blocks any URL that is explicitly blocked by a Deny URL, even if the same URL would otherwise be allowed by the Start URL check.

CLI users: To enable one or more actions, type "set appfw profile -denyURLaction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -denyURLaction none".

Default value: AS_DEFAULT_DISPOSITION

RefererHeaderCheck

Enable validation of Referer headers.

Referer validation ensures that a web form that a user sends to your web site originally came from your web site, not an outside attacker.

Although this parameter is part of the Start URL check, referer validation protects against cross-site request forgery (CSRF) attacks, not Start URL attacks.

Possible values: OFF, if_present, AlwaysExceptStartURLs, AlwaysExceptFirstRequest

Default value: OFF

cookieConsistencyAction

One or more Cookie Consistency actions. Available settings function as follows:

- * Block - Block connections that violate this security check.
- * Learn - Use the learning engine to generate a list of exceptions to this security check.
- * Log - Log violations of this security check.
- * Stats - Generate statistics for this security check.
- * None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -cookieConsistencyAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -cookieConsistencyAction none".

Default value: none

cookieTransforms

Perform the specified type of cookie transformation.

Available settings function as follows:

- * Encryption - Encrypt cookies.
- * Proxying - Mask contents of server cookies by sending proxy cookie to users.
- * Cookie flags - Flag cookies as HTTP only to prevent scripts on user's browser from accessing and possibly modifying them.

CAUTION: Make sure that this parameter is set to ON if you are configuring any cookie transformations. If it is set to OFF, no cookie transformations are performed regardless of any other settings.

Possible values: ON, OFF

Default value: OFF

cookieEncryption

Type of cookie encryption. Available settings function as follows:

- * None - Do not encrypt cookies.
- * Decrypt Only - Decrypt encrypted cookies, but do not encrypt cookies.
- * Encrypt Session Only - Encrypt session cookies, but not permanent cookies.
- * Encrypt All - Encrypt all cookies.

Possible values: none, decryptOnly, encryptSessionOnly, encryptAll

Default value: none

cookieProxying

Cookie proxy setting. Available settings function as follows:

- * None - Do not proxy cookies.
- * Session Only - Proxy session cookies by using the NetScaler session ID, but do not proxy permanent cookies.

Possible values: none, sessionOnly

Default value: none

addCookieFlags

Add the specified flags to cookies. Available settings function as follows:

- * None - Do not add flags to cookies.
- * HTTP Only - Add the HTTP Only flag to cookies, which prevents scripts from accessing cookies.
- * Secure - Add Secure flag to cookies.
- * All - Add both HTTPOnly and Secure flags to cookies.

Possible values: none, httpOnly, secure, all

Default value: none

fieldConsistencyAction

One or more Form Field Consistency actions. Available settings function as follows:

- * Block - Block connections that violate this security check.
- * Learn - Use the learning engine to generate a list of exceptions to this security check.
- * Log - Log violations of this security check.
- * Stats - Generate statistics for this security check.
- * None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -fieldConsistencyaction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -fieldConsistencyAction none".

Default value: none

CSRFtagAction

One or more Cross-Site Request Forgery (CSRF) Tagging actions. Available settings function as follows:

- * Block - Block connections that violate this security check.
- * Learn - Use the learning engine to generate a list of exceptions to this security check.
- * Log - Log violations of this security check.
- * Stats - Generate statistics for this security check.
- * None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -CSRFtagAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -CSRFtagAction none".

Default value: none

crossSiteScriptingAction

One or more Cross-Site Scripting (XSS) actions. Available settings function as follows:

- * Block - Block connections that violate this security check.
- * Learn - Use the learning engine to generate a list of exceptions to this security check.
- * Log - Log violations of this security check.
- * Stats - Generate statistics for this security check.
- * None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -crossSiteScriptingAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -crossSiteScriptingAction none".

Default value: AS_DEFAULT_DISPOSITION

crossSiteScriptingTransformUnsafeHTML

Transform cross-site scripts. This setting configures the application firewall to disable dangerous HTML instead of blocking the request.

CAUTION: Make sure that this parameter is set to ON if you are configuring any cross-site scripting transformations. If it is set to OFF, no cross-site scripting transformations are performed regardless of any other settings.

Possible values: ON, OFF

Default value: OFF

crossSiteScriptingCheckCompleteURLs

Check complete URLs for cross-site scripts, instead of just the query portions of URLs.

Possible values: ON, OFF

Default value: OFF

SQLInjectionAction

One or more HTML SQL Injection actions. Available settings function as follows:

- * Block - Block connections that violate this security check.
- * Learn - Use the learning engine to generate a list of exceptions to this security check.
- * Log - Log violations of this security check.
- * Stats - Generate statistics for this security check.
- * None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -SQLInjectionAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -SQLInjectionAction none".

Default value: AS_DEFAULT_DISPOSITION

SQLInjectionTransformSpecialChars

Transform injected SQL code. This setting configures the application firewall to disable SQL special strings instead of blocking the request. Since most SQL servers require a special string to activate an SQL keyword, in most cases a request that contains injected SQL code is safe if special strings are disabled.

CAUTION: Make sure that this parameter is set to ON if you are configuring any SQL injection transformations. If it is set to OFF, no SQL injection transformations are performed regardless of any other settings.

Possible values: ON, OFF

Default value: OFF

SQLInjectionType

Available SQL injection types.

- SQLSpIChar : Checks for SQL Special Chars
- SQLKeyword : Checks for SQL Keywords
- SQLSpICharANDKeyword : Checks for both and blocks if both are found
- SQLSpICharORKeyword : Checks for both and blocks if anyone is found

Possible values: SQLSplChar, SQLKeyword, SQLSplCharORKeyword, SQLSplCharANDKeyword

Default value: SQLSplCharANDKeyword

SQLInjectionCheckSQLWildChars

Check for form fields that contain SQL wild chars .

Possible values: ON, OFF

Default value: OFF

fieldFormatAction

One or more Field Format actions. Available settings function as follows:

- * Block - Block connections that violate this security check.
- * Learn - Use the learning engine to generate a list of suggested web form fields and field format assignments.
- * Log - Log violations of this security check.
- * Stats - Generate statistics for this security check.
- * None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -fieldFormatAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -fieldFormatAction none".

Default value: AS_DEFAULT_DISPOSITION

defaultFieldFormatType

Designate a default field type to be applied to web form fields that do not have a field type explicitly assigned to them.

defaultFieldFormatMinLength

Minimum length, in characters, for data entered into a field that is assigned the default field type.

To disable the minimum and maximum length settings and allow data of any length to be entered into the field, set this parameter to zero (0).

Default value: 0

Minimum value: 0

Maximum value: 65535

defaultFieldFormatMaxLength

Maximum length, in characters, for data entered into a field that is assigned the default field type.

Default value: 65535

Minimum value: 1

Maximum value: 65535

bufferOverflowAction

One or more Buffer Overflow actions. Available settings function as follows:

- * Block - Block connections that violate this security check.
- * Log - Log violations of this security check.
- * Stats - Generate statistics for this security check.
- * None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -bufferOverflowAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -bufferOverflowAction none".

Default value: AS_DEFAULT_DISPOSITION

bufferOverflowMaxURLLength

Maximum length, in characters, for URLs on your protected web sites. Requests with longer URLs are blocked.

Default value: 1024

Minimum value: 0

Maximum value: 65535

bufferOverflowMaxHeaderLength

Maximum length, in characters, for HTTP headers in requests sent to your protected web sites. Requests with longer headers are blocked.

Default value: 4096

Minimum value: 0

Maximum value: 65535

bufferOverflowMaxCookieLength

Maximum length, in characters, for cookies sent to your protected web sites. Requests with longer cookies are blocked.

Default value: 4096

Minimum value: 0

Maximum value: 65535

creditCardAction

One or more Credit Card actions. Available settings function as follows:

- * Block - Block connections that violate this security check.
- * Log - Log violations of this security check.
- * Stats - Generate statistics for this security check.
- * None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -creditCardAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -creditCardAction none".

Default value: none

creditCard

Credit card types that the application firewall should protect.

Default value: AS_CCARD_DEFAULT_CARD_TYPE

creditCardMaxAllowed

Maximum number of credit card numbers that can appear on a web page served by your protected web sites. Pages that contain more credit card numbers are blocked, or the credit card numbers are masked.

Minimum value: 0

Maximum value: 255

creditCardXOut

Mask any credit card number detected in a response by replacing each digit, except the digits in the final group, with the letter "X."

Possible values: ON, OFF

Default value: OFF

requestContentType

Default Content-Type header for requests.

A Content-Type header can contain 0-255 letters, numbers, and the hyphen (-) and underscore (_) characters.

Default value: NS_S_AS_DEFAULT_REQUEST_CONTENT_TYPE

responseContentType

Default Content-Type header for responses.

A Content-Type header can contain 0-255 letters, numbers, and the hyphen (-) and underscore (_) characters.

Default value: NS_S_AS_DEFAULT_RESPONSE_CONTENT_TYPE

XMLDoSAction

One or more XML Denial-of-Service (XDoS) actions. Available settings function as follows:

- * Block - Block connections that violate this security check.
- * Learn - Use the learning engine to generate a list of exceptions to this security check.
- * Log - Log violations of this security check.
- * Stats - Generate statistics for this security check.
- * None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -XMLDoSAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -XMLDoSAction none".

Default value: AS_DEFAULT_DISPOSITION

XMLFormatAction

One or more XML Format actions. Available settings function as follows:

- * Block - Block connections that violate this security check.
- * Log - Log violations of this security check.
- * Stats - Generate statistics for this security check.
- * None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -XMLFormatAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -XMLFormatAction none".

Default value: AS_DEFAULT_DISPOSITION

XMLSQLInjectionAction

One or more XML SQL Injection actions. Available settings function as follows:

- * Block - Block connections that violate this security check.
- * Log - Log violations of this security check.
- * Stats - Generate statistics for this security check.

* None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -XMLSQLInjectionAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -XMLSQLInjectionAction none".

Default value: AS_DEFAULT_DISPOSITION

XMLSQLInjectionType

Available SQL injection types.

-SQLSpIChar : Checks for SQL Special Chars

-SQLKeyword : Checks for SQL Keywords

-SQLSpICharANDKeyword : Checks for both and blocks if both are found

-SQLSpICharORKeyword : Checks for both and blocks if anyone is found

Possible values: SQLSpIChar, SQLKeyword, SQLSpICharORKeyword, SQLSpICharANDKeyword

Default value: SQLSpICharANDKeyword

XMLSQLInjectionCheckSQLWildChars

Check for form fields that contain SQL wild chars .

Possible values: ON, OFF

Default value: OFF

XMLSQLInjectionParseComments

Parse comments in XML Data and exempt those sections of the request that are from the XML SQL Injection check. You must configure the type of comments that the application firewall is to detect and exempt from this security check. Available settings function as follows:

* Check all - Check all content.

* ANSI - Exempt content that is part of an ANSI (Mozilla-style) comment.

* Nested - Exempt content that is part of a nested (Microsoft-style) comment.

* ANSI Nested - Exempt content that is part of any type of comment.

Possible values: checkall, ansi, nested, ansinested

Default value: checkall

XMLXSSAction

One or more XML Cross-Site Scripting actions. Available settings function as follows:

* Block - Block connections that violate this security check.

* Log - Log violations of this security check.

* Stats - Generate statistics for this security check.

* None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -XMLXSSAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -XMLXSSAction none".

Default value: AS_DEFAULT_DISPOSITION

XMLWSIAction

One or more Web Services Interoperability (WSI) actions. Available settings function as follows:

* Block - Block connections that violate this security check.

- * Learn - Use the learning engine to generate a list of exceptions to this security check.
- * Log - Log violations of this security check.
- * Stats - Generate statistics for this security check.
- * None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -XMLWSIAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -XMLWSIAction none".

Default value: AS_DEFAULT_DISPOSITION

XMLAttachmentAction

One or more XML Attachment actions. Available settings function as follows:

- * Block - Block connections that violate this security check.
- * Learn - Use the learning engine to generate a list of exceptions to this security check.
- * Log - Log violations of this security check.
- * Stats - Generate statistics for this security check.
- * None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -XMLAttachmentAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -XMLAttachmentAction none".

Default value: AS_DEFAULT_DISPOSITION

XMLValidationAction

One or more XML Validation actions. Available settings function as follows:

- * Block - Block connections that violate this security check.
- * Log - Log violations of this security check.
- * Stats - Generate statistics for this security check.
- * None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -XMLValidationAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -XMLValidationAction none".

Default value: AS_DEFAULT_DISPOSITION

XMLErrorObject

Name to assign to the XML Error Object, which the application firewall displays when a user request is blocked.

Must begin with a letter, number, or the underscore character \(_\), and must contain only letters, numbers, and the hyphen \(-\), period \(\.\), pound \(\#\), space \(\), at \(@\), equals \(\=\), colon \(\:\), and underscore characters. Cannot be changed after the XML error object is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks \(''\) (for example, "my XML error object" or 'my XML error object').

Default value: NS_S_AS_ERROR_OBJECT_DEFAULT

signatures

Object name for signatures.

This check is applicable to Profile Type: HTML, XML.

Default value: NS_S_AS_CUSTOM_OBJECT_DEFAULT

XMLSOAPFaultAction

One or more XML SOAP Fault Filtering actions. Available settings function as follows:

- * Block - Block connections that violate this security check.
- * Log - Log violations of this security check.
- * Stats - Generate statistics for this security check.
- * None - Disable all actions for this security check.
- * Remove - Remove all violations for this security check.

CLI users: To enable one or more actions, type "set appfw profile -XMLSOAPFaultAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -XMLSOAPFaultAction none".

Default value: AS_DEFAULT_DISPOSITION

useHTMLErrorObject

Send an imported HTML Error object to a user when a request is blocked, instead of redirecting the user to the designated Error URL.

Possible values: ON, OFF

Default value: OFF

errorURL

URL that application firewall uses as the Error URL.

Default value: NS_S_AS_ERROR_URL_DEFAULT

HTMLErrorObject

Name to assign to the HTML Error Object.

Must begin with a letter, number, or the underscore character `\\(_\\)`, and must contain only letters, numbers, and the hyphen `\\(-\\)`, period `\\(.\\)`, pound `\\(\\#\\)`, space `\\(\\)`, at `\\(@\\)`, equals `\\(=\\)`, colon `\\(:\\)`, and underscore characters. Cannot be changed after the HTML error object is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks `\\(`for example, "my HTML error object" or 'my HTML error object'`\\)`.

Default value: NS_S_AS_ERROR_OBJECT_DEFAULT

logEveryPolicyHit

Log every profile match, regardless of security checks results.

Possible values: ON, OFF

Default value: OFF

stripHtmlComments

Strip HTML comments before forwarding a web page sent by a protected web site in response to a user request.

Possible values: none, all, exclude_script_tag

Default value: none

stripXmlComments

Exempt URLs that pass the Start URL closure check from additional security checks.

Possible values: none, all

Default value: none

exemptClosureURLsFromSecurityChecks

Exempt URLs that pass the Start URL closure check from additional security checks.

Possible values: ON, OFF

Default value: ON

defaultCharSet

Default character set for protected web pages. Web pages sent by your protected web sites in response to user requests are assigned this character set if the page does not already specify a character set. The character sets supported by the application firewall are:

* iso-8859-1 (English US)

* big5 (Chinese Traditional)

* gb2312 (Chinese Simplified)

* sjis (Japanese Shift-JIS)

* euc-jp (Japanese EUC-JP)

* iso-8859-9 (Turkish)

* utf-8 (Unicode)

* euc-kr (Korean)

Default value: NS_S_AS_CHARSET_DEFAULT

Maximum value: 31

postBodyLimit

Maximum allowed HTTP post body size, in bytes.

Default value: 20000000

Minimum value: 0

Maximum value: 1000000000

fileUploadMaxNum

Maximum allowed number of file uploads per form-submission request. The maximum setting (65535) allows an unlimited number of uploads.

Default value: 65535

Minimum value: 0

Maximum value: 65535

canonicalizeHTMLResponse

Perform HTML entity encoding for any special characters in responses sent by your protected web sites.

Possible values: ON, OFF

Default value: ON

enableFormTagging

Enable tagging of web form fields for use by the Form Field Consistency and CSRF Form Tagging checks.

Possible values: ON, OFF

Default value: ON

sessionlessFieldConsistency

Perform sessionless Field Consistency Checks.

Possible values: OFF, ON, postOnly

Default value: OFF

sessionlessURLClosure

Enable session less URL Closure Checks.

This check is applicable to Profile Type: HTML.

Possible values: ON, OFF

Default value: OFF

semicolonFieldSeparator

Allow ';' as a form field separator in URL queries and POST form bodies.

Possible values: ON, OFF

Default value: OFF

excludeFileUploadFromChecks

Exclude uploaded files from Form checks.

Possible values: ON, OFF

Default value: OFF

SQLInjectionParseComments

Parse HTML comments and exempt them from the HTML SQL Injection check. You must specify the type of comments that the application firewall is to detect and exempt from this security check. Available settings function as follows:

- * Check all - Check all content.
- * ANSI - Exempt content that is part of an ANSI (Mozilla-style) comment.
- * Nested - Exempt content that is part of a nested (Microsoft-style) comment.
- * ANSI Nested - Exempt content that is part of any type of comment.

Possible values: checkall, ansi, nested, ansinested

Default value: AS_DEFAULT_SQLINJECTIONPARSECOMMENTS

invalidPercentHandling

Configure the method that the application firewall uses to handle percent-encoded names and values. Available settings function as follows:

- * apache_mode - Apache format.
- * asp_mode - Microsoft ASP format.
- * secure_mode - Secure format.

Possible values: apache_mode, asp_mode, secure_mode

Default value: secure_mode

type

Application firewall profile type, which controls which security checks and settings are applied to content that is filtered with the profile. Available settings function as follows:

- * HTML - HTML-based web sites.

- * XML - XML-based web sites and services.

- * HTML XML (Web 2.0) - Sites that contain both HTML and XML content, such as ATOM feeds, blogs, and RSS feeds.

Default value: HTML

checkRequestHeaders

Check request headers as well as web forms for injected SQL and cross-site scripts.

Possible values: ON, OFF

Default value: OFF

optimizePartialReqs

Optimize handle of HTTP partial requests i.e. those with range headers.

Available settings are as follows:

- * ON - Partial requests by the client result in partial requests to the backend server in most cases.

- * OFF - Partial requests by the client are changed to full requests to the backend server

Possible values: ON, OFF

Default value: ON

URLDecodeRequestCookies

URL Decode request cookies before subjecting them to SQL and cross-site scripting checks.

Possible values: ON, OFF

Default value: OFF

comment

Any comments about the purpose of profile, or other useful information about the profile.

rm appfw profile

Removes the specified application firewall profile.

Synopsys

rm appfw profile <name>

Arguments

name

Name of the profile.

set appfw profile

Modifies the specified parameters of the specified application firewall profile.

Synopsys

set appfw profile <name> [-startURLAction <startURLAction> ...] [-contentTypeAction <contentTypeAction> ...] [-startURLClosure (ON | OFF)] [-denyURLAction <denyURLAction> ...] [-RefererHeaderCheck <RefererHeaderCheck>] [-cookieConsistencyAction <cookieConsistencyAction> ...] [-cookieTransforms (ON | OFF)] [-cookieEncryption <cookieEncryption>] [-cookieProxying (none | sessionOnly)] [-addCookieFlags <addCookieFlags>] [-fieldConsistencyAction <fieldConsistencyAction> ...] [-CSRFtagAction <CSRFtagAction> ...] [-crossSiteScriptingAction <crossSiteScriptingAction> ...] [-crossSiteScriptingTransformUnsafeHTML (ON | OFF)] [-crossSiteScriptingCheckCompleteURLs (ON | OFF)] [-SQLInjectionAction <SQLInjectionAction> ...] [-SQLInjectionTransformSpecialChars (ON | OFF)] [-SQLInjectionType <SQLInjectionType>] [-SQLInjectionCheckSQLWildChars (ON | OFF)] [-fieldFormatAction <fieldFormatAction> ...] [-defaultFieldFormatType <string>] [-defaultFieldFormatMinLength <positive_integer>] [-defaultFieldFormatMaxLength <positive_integer>] [-bufferOverflowAction <bufferOverflowAction> ...] [-bufferOverflowMaxURLLength <positive_integer>] [-bufferOverflowMaxHeaderLength <positive_integer>] [-bufferOverflowMaxCookieLength <positive_integer>] [-creditCardAction <creditCardAction> ...] [-creditCard <creditCard> ...] [-creditCardMaxAllowed <positive_integer>] [-creditCardXOut (ON | OFF)] [-requestContentType <string>] [-responseContentType <string>] [-XMLDoSAction <XMLDoSAction> ...] [-XMLFormatAction <XMLFormatAction> ...] [-XMLSQLInjectionAction <XMLSQLInjectionAction> ...] [-XMLSQLInjectionType <XMLSQLInjectionType>] [-XMLSQLInjectionCheckSQLWildChars (ON | OFF)] [-XMLSQLInjectionParseComments <XMLSQLInjectionParseComments>] [-XMLXSSAction <XMLXSSAction> ...] [-XMLWSIAction <XMLWSIAction> ...] [-XMLAttachmentAction <XMLAttachmentAction> ...] [-XMLValidationAction <XMLValidationAction> ...] [-XMLErrorObject <string>] [-signatures <string>] [-XMLSOAPFaultAction <XMLSOAPFaultAction> ...] [-useHTMLErrorObject (ON | OFF)] [-errorURL <expression>] [-HTMLErrorObject <string>] [-logEveryPolicyHit (ON | OFF)] [-stripHtmlComments <stripHtmlComments>] [-stripXmlComments (none | all)] [-exemptClosureURLsFromSecurityChecks (ON | OFF)] [-defaultCharSet <string>] [-postBodyLimit <positive_integer>] [-fileUploadMaxNum <positive_integer>] [-canonicalizeHTMLResponse (ON | OFF)] [-enableFormTagging (ON | OFF)] [-sessionlessFieldConsistency <sessionlessFieldConsistency>] [-sessionlessURLClosure (ON | OFF)] [-semicolonFieldSeparator (ON | OFF)] [-excludeFileUploadFromChecks (ON | OFF)] [-SQLInjectionParseComments <SQLInjectionParseComments>] [-invalidPercentHandling <invalidPercentHandling>] [-type (HTML | XML) ...] [-checkRequestHeaders (ON | OFF)] [-optimizePartialReqs (ON | OFF)] [-URLDecodeRequestCookies (ON | OFF)] [-comment <string>]

Arguments

name

Name of the profile that you want to modify.

startURLAction

One or more Start URL actions. Available settings function as follows:

- * Block - Block connections that violate this security check.
- * Learn - Use the learning engine to generate a list of exceptions to this security check.
- * Log - Log violations of this security check.
- * Stats - Generate statistics for this security check.
- * None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -startURLAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -startURLAction none".

Default value: AS_DEFAULT_DISPOSITION

contentTypeAction

One or more Content-type actions. Available settings function as follows:

- * Block - Block connections that violate this security check.
- * Log - Log violations of this security check.
- * None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -contentTypeaction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -contentTypeaction none".

Default value: AS_DEFAULT_CONTENT_TYPE_DISPOSITION

startURLClosure

Toggle the state of Start URL Closure.

Possible values: ON, OFF

Default value: OFF

denyURLAction

One or more Deny URL actions. Available settings function as follows:

- * Block - Block connections that violate this security check.
- * Log - Log violations of this security check.
- * Stats - Generate statistics for this security check.
- * None - Disable all actions for this security check.

NOTE: The Deny URL check takes precedence over the Start URL check. If you enable blocking for the Deny URL check, the application firewall blocks any URL that is explicitly blocked by a Deny URL, even if the same URL would otherwise be allowed by the Start URL check.

CLI users: To enable one or more actions, type "set appfw profile -denyURLaction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -denyURLaction none".

Default value: AS_DEFAULT_DISPOSITION

RefererHeaderCheck

Enable validation of Referer headers.

Referer validation ensures that a web form that a user sends to your web site originally came from your web site, not an outside attacker.

Although this parameter is part of the Start URL check, referer validation protects against cross-site request forgery (CSRF) attacks, not Start URL attacks.

Possible values: OFF, if_present, AlwaysExceptStartURLs, AlwaysExceptFirstRequest

Default value: OFF

cookieConsistencyAction

One or more Cookie Consistency actions. Available settings function as follows:

- * Block - Block connections that violate this security check.
- * Learn - Use the learning engine to generate a list of exceptions to this security check.
- * Log - Log violations of this security check.
- * Stats - Generate statistics for this security check.
- * None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -cookieConsistencyAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -cookieConsistencyAction none".

Default value: none

cookieTransforms

Perform the specified type of cookie transformation.

Available settings function as follows:

- * Encryption - Encrypt cookies.
- * Proxying - Mask contents of server cookies by sending proxy cookie to users.

* Cookie flags - Flag cookies as HTTP only to prevent scripts on user's browser from accessing and possibly modifying them.

CAUTION: Make sure that this parameter is set to ON if you are configuring any cookie transformations. If it is set to OFF, no cookie transformations are performed regardless of any other settings.

Possible values: ON, OFF

cookieEncryption

Type of cookie encryption. Available settings function as follows:

- * None - Do not encrypt cookies.
- * Decrypt Only - Decrypt encrypted cookies, but do not encrypt cookies.
- * Encrypt Session Only - Encrypt session cookies, but not permanent cookies.
- * Encrypt All - Encrypt all cookies.

Possible values: none, decryptOnly, encryptSessionOnly, encryptAll

Default value: none

cookieProxying

Cookie proxy setting. Available settings function as follows:

- * None - Do not proxy cookies.
- * Session Only - Proxy session cookies by using the NetScaler session ID, but do not proxy permanent cookies.

Possible values: none, sessionOnly

Default value: none

addCookieFlags

Add HttpOnly and Secure flags to cookies

Possible values: none, httpOnly, secure, all

Default value: none

fieldConsistencyAction

One or more Form Field Consistency actions. Available settings function as follows:

- * Block - Block connections that violate this security check.
- * Learn - Use the learning engine to generate a list of exceptions to this security check.
- * Log - Log violations of this security check.
- * Stats - Generate statistics for this security check.
- * None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -fieldConsistencyaction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -fieldConsistencyAction none".

Default value: none

CSRFtagAction

One or more Cross-Site Request Forgery (CSRF) Tagging actions. Available settings function as follows:

- * Block - Block connections that violate this security check.
- * Learn - Use the learning engine to generate a list of exceptions to this security check.

- * Log - Log violations of this security check.
- * Stats - Generate statistics for this security check.
- * None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -CSRFTagAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -CSRFTagAction none".

Default value: none

crossSiteScriptingAction

One or more Cross-Site Scripting (XSS) actions. Available settings function as follows:

- * Block - Block connections that violate this security check.
- * Learn - Use the learning engine to generate a list of exceptions to this security check.
- * Log - Log violations of this security check.
- * Stats - Generate statistics for this security check.
- * None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -crossSiteScriptingAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -crossSiteScriptingAction none".

Default value: AS_DEFAULT_DISPOSITION

crossSiteScriptingTransformUnsafeHTML

Transform cross-site scripts. This setting configures the application firewall to disable dangerous HTML instead of blocking the request.

CAUTION: Make sure that this parameter is set to ON if you are configuring any cross-site scripting transformations. If it is set to OFF, no cross-site scripting transformations are performed regardless of any other settings.

Possible values: ON, OFF

crossSiteScriptingCheckCompleteURLs

Check complete URLs for cross-site scripts, instead of just the query portions of URLs.

Possible values: ON, OFF

SQLInjectionAction

One or more HTML SQL Injection actions. Available settings function as follows:

- * Block - Block connections that violate this security check.
- * Learn - Use the learning engine to generate a list of exceptions to this security check.
- * Log - Log violations of this security check.
- * Stats - Generate statistics for this security check.
- * None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -SQLInjectionAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -SQLInjectionAction none".

Default value: AS_DEFAULT_DISPOSITION

SQLInjectionTransformSpecialChars

Transform injected SQL code. This setting configures the application firewall to disable SQL special strings instead of blocking the request. Since most SQL servers require a special string to activate an SQL keyword, in most cases a request that contains injected SQL code is safe if special strings are disabled.

CAUTION: Make sure that this parameter is set to ON if you are configuring any SQL injection transformations. If it is set to OFF, no SQL injection transformations are performed regardless of any other settings.

Possible values: ON, OFF

SQLInjectionType

Available SQL injection types.

-SQLSpIChar : Checks for SQL Special Chars

-SQLKeyword : Checks for SQL Keywords

-SQLSpICharANDKeyword : Checks for both and blocks if both are found

-SQLSpICharORKeyword : Checks for both and blocks if anyone is found

Possible values: SQLSpIChar, SQLKeyword, SQLSpICharORKeyword, SQLSpICharANDKeyword

SQLInjectionCheckSQLWildChars

Check for form fields that contain SQL wild chars .

Possible values: ON, OFF

fieldFormatAction

One or more Field Format actions. Available settings function as follows:

* Block - Block connections that violate this security check.

* Learn - Use the learning engine to generate a list of suggested web form fields and field format assignments.

* Log - Log violations of this security check.

* Stats - Generate statistics for this security check.

* None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -fieldFormatAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -fieldFormatAction none".

Default value: AS_DEFAULT_DISPOSITION

defaultFieldFormatType

Designate a default field type to be applied to web form fields that do not have a field type explicitly assigned to them.

defaultFieldFormatMinLength

Minimum length, in characters, for data entered into a field that is assigned the default field type.

To disable the minimum and maximum length settings and allow data of any length to be entered into the field, set this parameter to zero (0).

Default value: 0

Minimum value: 0

Maximum value: 65535

defaultFieldFormatMaxLength

Maximum length, in characters, for data entered into a field that is assigned the default field type.

Default value: 65535

Minimum value: 1

Maximum value: 65535

bufferOverflowAction

One or more Buffer Overflow actions. Available settings function as follows:

- * Block - Block connections that violate this security check.
- * Log - Log violations of this security check.
- * Stats - Generate statistics for this security check.
- * None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -bufferOverflowAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -bufferOverflowAction none".

Default value: AS_DEFAULT_DISPOSITION

bufferOverflowMaxURLLength

Maximum length, in characters, for URLs on your protected web sites. Requests with longer URLs are blocked.

Default value: 1024

Minimum value: 0

Maximum value: 65535

bufferOverflowMaxHeaderLength

Maximum length, in characters, for HTTP headers in requests sent to your protected web sites. Requests with longer headers are blocked.

Default value: 4096

Minimum value: 0

Maximum value: 65535

bufferOverflowMaxCookieLength

Maximum length, in characters, for cookies sent to your protected web sites. Requests with longer cookies are blocked.

Default value: 4096

Minimum value: 0

Maximum value: 65535

creditCardAction

One or more Credit Card actions. Available settings function as follows:

- * Block - Block connections that violate this security check.
- * Log - Log violations of this security check.
- * Stats - Generate statistics for this security check.
- * None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -creditCardAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -creditCardAction none".

Default value: none

creditCard

Credit card types that the application firewall should protect.

Default value: AS_CCARD_DEFAULT_CARD_TYPE

creditCardMaxAllowed

Maximum number of credit card numbers that can appear on a web page served by your protected web sites. Pages that contain more credit card numbers are blocked, or the credit card numbers are masked.

Minimum value: 0

Maximum value: 255

creditCardXOut

Mask any credit card number detected in a response by replacing each digit, except the digits in the final group, with the letter "X."

Possible values: ON, OFF

requestContentType

Default Content-Type header for requests.

A Content-Type header can contain 0-255 letters, numbers, and the hyphen (-) and underscore (_) characters.

Default value: NS_S_AS_DEFAULT_REQUEST_CONTENT_TYPE

responseContentType

Default Content-Type header for responses.

A Content-Type header can contain 0-255 letters, numbers, and the hyphen (-) and underscore (_) characters.

Default value: NS_S_AS_DEFAULT_RESPONSE_CONTENT_TYPE

XMLDoSAction

One or more XML Denial-of-Service (XDoS) actions. Available settings function as follows:

- * Block - Block connections that violate this security check.
- * Learn - Use the learning engine to generate a list of exceptions to this security check.
- * Log - Log violations of this security check.
- * Stats - Generate statistics for this security check.
- * None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -XMLDoSAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -XMLDoSAction none".

Default value: AS_DEFAULT_DISPOSITION

XMLFormatAction

One or more XML Format actions. Available settings function as follows:

- * Block - Block connections that violate this security check.
- * Log - Log violations of this security check.
- * Stats - Generate statistics for this security check.
- * None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -XMLFormatAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -XMLFormatAction none".

Default value: AS_DEFAULT_DISPOSITION

XMLSQLInjectionAction

One or more XML SQL Injection actions. Available settings function as follows:

- * Block - Block connections that violate this security check.
- * Log - Log violations of this security check.
- * Stats - Generate statistics for this security check.
- * None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -XMLSQLInjectionAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -XMLSQLInjectionAction none".

Default value: AS_DEFAULT_DISPOSITION

XMLSQLInjectionType

Available SQL injection types.

- SQLSpIChar : Checks for SQL Special Chars
- SQLKeyword : Checks for SQL Keywords
- SQLSpICharANDKeyword : Checks for both and blocks if both are found
- SQLSpICharORKeyword : Checks for both and blocks if anyone is found

Possible values: SQLSpIChar, SQLKeyword, SQLSpICharORKeyword, SQLSpICharANDKeyword

XMLSQLInjectionCheckSQLWildChars

Check for form fields that contain SQL wild chars .

Possible values: ON, OFF

XMLSQLInjectionParseComments

Parse comments in XML Data and exempt those sections of the request that are from the XML SQL Injection check. You must configure the type of comments that the application firewall is to detect and exempt from this security check. Available settings function as follows:

- * Check all - Check all content.
- * ANSI - Exempt content that is part of an ANSI (Mozilla-style) comment.
- * Nested - Exempt content that is part of a nested (Microsoft-style) comment.
- * ANSI Nested - Exempt content that is part of any type of comment.

Possible values: checkall, ansi, nested, ansinested

Default value: checkall

XMLXSSAction

One or more XML Cross-Site Scripting actions. Available settings function as follows:

- * Block - Block connections that violate this security check.
- * Log - Log violations of this security check.
- * Stats - Generate statistics for this security check.
- * None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -XMLXSSAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -XMLXSSAction none".

Default value: AS_DEFAULT_DISPOSITION

XMLWSIAction

One or more Web Services Interoperability (WSI) actions. Available settings function as follows:

- * Block - Block connections that violate this security check.
- * Learn - Use the learning engine to generate a list of exceptions to this security check.
- * Log - Log violations of this security check.
- * Stats - Generate statistics for this security check.
- * None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -XMLWSIAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -XMLWSIAction none".

Default value: AS_DEFAULT_DISPOSITION

XMLAttachmentAction

One or more XML Attachment actions. Available settings function as follows:

- * Block - Block connections that violate this security check.
- * Learn - Use the learning engine to generate a list of exceptions to this security check.
- * Log - Log violations of this security check.
- * Stats - Generate statistics for this security check.
- * None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -XMLAttachmentAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -XMLAttachmentAction none".

Default value: AS_DEFAULT_DISPOSITION

XMLValidationAction

One or more XML Validation actions. Available settings function as follows:

- * Block - Block connections that violate this security check.
- * Log - Log violations of this security check.
- * Stats - Generate statistics for this security check.
- * None - Disable all actions for this security check.

CLI users: To enable one or more actions, type "set appfw profile -XMLValidationAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -XMLValidationAction none".

Default value: AS_DEFAULT_DISPOSITION

XMLErrorObject

Name to assign to the XML Error Object, which the application firewall displays when a user request is blocked.

Must begin with a letter, number, or the underscore character _\\, and must contain only letters, numbers, and the hyphen \\(-\\), period \\(.\\) pound \\(\\#\\), space \\(\\), at (@), equals \\(=\\), colon \\(:\\), and underscore characters. Cannot be changed after the XML error object is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks \\(for example, "my XML error object" or 'my XML error object\\).

Default value: NS_S_AS_ERROR_OBJECT_DEFAULT

signatures

Object name for signatures.

This check is applicable to Profile Type: HTML, XML.

Default value: NS_S_AS_CUSTOM_OBJECT_DEFAULT

XMLSOAPFaultAction

One or more XML SOAP Fault Filtering actions. Available settings function as follows:

- * Block - Block connections that violate this security check.
- * Log - Log violations of this security check.
- * Stats - Generate statistics for this security check.
- * None - Disable all actions for this security check.
- * Remove - Remove all violations for this security check.

CLI users: To enable one or more actions, type "set appfw profile -XMLSOAPFaultAction" followed by the actions to be enabled. To turn off all actions, type "set appfw profile -XMLSOAPFaultAction none".

Default value: AS_DEFAULT_DISPOSITION

useHTMLErrorObject

Send an imported HTML Error object to a user when a request is blocked, instead of redirecting the user to the designated Error URL.

Possible values: ON, OFF

errorURL

URL that application firewall uses as the Error URL.

Default value: NS_S_AS_ERROR_URL_DEFAULT

HTMLErrorObject

Name to assign to the HTML Error Object.

Must begin with a letter, number, or the underscore character \(_\), and must contain only letters, numbers, and the hyphen \(-\), period \(\.\) pound \(\#\), space \(\), at (@), equals \(\=\), colon \(\:\), and underscore characters. Cannot be changed after the HTML error object is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks \("for example, "my HTML error object" or 'my HTML error object'\).

Default value: NS_S_AS_ERROR_OBJECT_DEFAULT

logEveryPolicyHit

Log every profile match, regardless of security checks results.

Possible values: ON, OFF

stripHtmlComments

Strip HTML comments before forwarding a web page sent by a protected web site in response to a user request.

Possible values: none, all, exclude_script_tag

stripXmlComments

Exempt URLs that pass the Start URL closure check from additional security checks.

Possible values: none, all

exemptClosureURLsFromSecurityChecks

Exempt URLs that pass the Start URL closure check from additional security checks.

Possible values: ON, OFF

defaultCharSet

Default character set for protected web pages. Web pages sent by your protected web sites in response to user requests are assigned this character set if the page does not already specify a character set. The character sets supported by the application firewall are:

* iso-8859-1 (English US)

* big5 (Chinese Traditional)

* gb2312 (Chinese Simplified)

* sjis (Japanese Shift-JIS)

* euc-jp (Japanese EUC-JP)

* iso-8859-9 (Turkish)

* utf-8 (Unicode)

* euc-kr (Korean)

Default value: NS_S_AS_CHARSET_DEFAULT

Maximum value: 31

postBodyLimit

Maximum allowed HTTP post body size, in bytes.

Default value: 20000000

Minimum value: 0

Maximum value: 1000000000

fileUploadMaxNum

Maximum allowed number of file uploads per form-submission request. The maximum setting (65535) allows an unlimited number of uploads.

Default value: 65535

Minimum value: 0

Maximum value: 65535

canonicalizeHTMLResponse

Perform HTML entity encoding for any special characters in responses sent by your protected web sites.

Possible values: ON, OFF

Default value: ON

enableFormTagging

Enable tagging of web form fields for use by the Form Field Consistency and CSRF Form Tagging checks.

Possible values: ON, OFF

Default value: ON

sessionlessFieldConsistency

Perform sessionless Field Consistency Checks.

Possible values: OFF, ON, postOnly

Default value: OFF

sessionlessURLClosure

Enable session less URL Closure Checks.

This check is applicable to Profile Type: HTML.

Possible values: ON, OFF

Default value: OFF

semicolonFieldSeparator

Allow ';' as a form field separator in URL queries and POST form bodies.

Possible values: ON, OFF

Default value: OFF

excludeFileUploadFromChecks

Exclude uploaded files from Form checks.

Possible values: ON, OFF

Default value: OFF

SQLInjectionParseComments

Parse HTML comments and exempt them from the HTML SQL Injection check. You must specify the type of comments that the application firewall is to detect and exempt from this security check. Available settings function as follows:

- * Check all - Check all content.
- * ANSI - Exempt content that is part of an ANSI (Mozilla-style) comment.
- * Nested - Exempt content that is part of a nested (Microsoft-style) comment.
- * ANSI Nested - Exempt content that is part of any type of comment.

Possible values: checkall, ansi, nested, ansinested

Default value: AS_DEFAULT_SQLINJECTIONPARSECOMMENTS

invalidPercentHandling

Configure the method that the application firewall uses to handle percent-encoded names and values. Available settings function as follows:

- * apache_mode - Apache format.
- * asp_mode - Microsoft ASP format.
- * secure_mode - Secure format.

Possible values: apache_mode, asp_mode, secure_mode

Default value: secure_mode

type

Application firewall profile type, which controls which security checks and settings are applied to content that is filtered with the profile. Available settings function as follows:

- * HTML - HTML-based web sites.

- * XML - XML-based web sites and services.

- * HTML XML (Web 2.0) - Sites that contain both HTML and XML content, such as ATOM feeds, blogs, and RSS feeds.

Default value: HTML

checkRequestHeaders

Check request headers as well as web forms for injected SQL and cross-site scripts.

Possible values: ON, OFF

Default value: OFF

optimizePartialReqs

Optimize handle of HTTP partial requests i.e. those with range headers.

Available settings are as follows:

- * ON - Partial requests by the client result in partial requests to the backend server in most cases.

- * OFF - Partial requests by the client are changed to full requests to the backend server

Possible values: ON, OFF

URLDecodeRequestCookies

URL Decode request cookies before subjecting them to SQL and cross-site scripting checks.

Possible values: ON, OFF

Default value: OFF

comment

Any comments about the purpose of profile, or other useful information about the profile.

unset appfw profile

Use this command to remove appfw profile settings. Refer to the set appfw profile command for meanings of the arguments.

Synopsys

```
unset appfw profile <name> [-startURLAction] [-contentTypeAction] [-startURLClosure] [-denyURLAction] [-
RefererHeaderCheck] [-cookieConsistencyAction] [-cookieTransforms] [-cookieEncryption] [-cookieProxying] [-
addCookieFlags] [-fieldConsistencyAction] [-CSRFtagAction] [-crossSiteScriptingAction] [-
crossSiteScriptingTransformUnsafeHTML] [-crossSiteScriptingCheckCompleteURLs] [-SQLInjectionAction] [-
SQLInjectionTransformSpecialChars] [-SQLInjectionType] [-SQLInjectionCheckSQLWildChars] [-fieldFormatAction] [-
defaultFieldFormatType] [-defaultFieldFormatMinLength] [-defaultFieldFormatMaxLength] [-bufferOverflowAction] [-
bufferOverflowMaxURLLength] [-bufferOverflowMaxHeaderLength] [-bufferOverflowMaxCookieLength] [-
creditCardAction] [-creditCard] [-creditCardMaxAllowed] [-creditCardXOut] [-requestContentType] [-
responseContentType] [-XMLDoSAction] [-XMLFormatAction] [-XMLSQLInjectionAction] [-XMLSQLInjectionType] [-
XMLSQLInjectionCheckSQLWildChars] [-XMLSQLInjectionParseComments] [-XMLXSSAction] [-XMLWSIAction] [-
XMLAttachmentAction] [-XMLValidationAction] [-XMLErrorObject] [-signatures] [-XMLSOAPFaultAction] [-
useHTMLErrorObject] [-errorURL] [-HTMLErrorObject] [-logEveryPolicyHit] [-stripHtmlComments] [-
stripXmlComments] [-exemptClosureURLsFromSecurityChecks] [-defaultCharSet] [-postBodyLimit] [-
fileUploadMaxNum] [-canonicalizeHTMLResponse] [-enableFormTagging] [-sessionlessFieldConsistency] [-
sessionlessURLClosure] [-semicolonFieldSeparator] [-excludeFileUploadFromChecks] [-
SQLInjectionParseComments] [-invalidPercentHandling] [-type] [-checkRequestHeaders] [-optimizePartialReqs] [-
URLDecodeRequestCookies] [-comment]
```

bind appfw profile

Binds the specified exemption (relaxation) or rule to the specified application firewall profile. NOTE: You should not attempt to bind more than one exemption or rule at a time by using this command.

Synopsys

```
bind appfw profile <name> (-startURL <expression> | (-denyURL <expression> | (-fieldConsistency <string>
<formActionURL> [-isRegex ( REGEX | NOTREGEX )]) | (-cookieConsistency <string> [-isRegex ( REGEX |
NOTREGEX )]) | (-SQLInjection <string> <formActionURL> [-isRegex ( REGEX | NOTREGEX )]) [-location
<location>]) | (-CSRFTag <expression> <CSRFFormActionURL>) | (-crossSiteScripting <string> <formActionURL> [-
isRegex ( REGEX | NOTREGEX )]) [-location <location>]) | (-fieldFormat <string> <formActionURL> <fieldType> [-
fieldFormatMinLength <positive_integer>] [-fieldFormatMaxLength <positive_integer>] [-isRegex ( REGEX |
NOTREGEX )]) | (-safeObject <string> <expression> <maxMatchLength> [-action <action> ...]) | -
trustedLearningClients <ip_addr[/prefix]]ip6_addr[/prefix]]*> | (-XMLDoSURL <expression> [-
XMLMaxElementDepthCheck ( ON | OFF ) [-XMLMaxElementDepth <positive_integer>]] [-
XMLMaxElementNameLengthCheck ( ON | OFF ) [-XMLMaxElementNameLength <positive_integer>]] [-
XMLMaxElementsCheck ( ON | OFF ) [-XMLMaxElements <positive_integer>]] [-XMLMaxElementChildrenCheck (
ON | OFF ) [-XMLMaxElementChildren <positive_integer>]] [-XMLMaxAttributesCheck ( ON | OFF ) [-
XMLMaxAttributes <positive_integer>]] [-XMLMaxAttributeNameLengthCheck ( ON | OFF ) [-
XMLMaxAttributeNameLength <positive_integer>]] [-XMLMaxAttributeValueLengthCheck ( ON | OFF ) [-
XMLMaxAttributeValueLength <positive_integer>]] [-XMLMaxCharDATALengthCheck ( ON | OFF ) [-
XMLMaxCharDATALength <positive_integer>]] [-XMLMaxFileSizeCheck ( ON | OFF ) [-XMLMaxFileSize
<positive_integer>]] [-XMLMinFileSizeCheck ( ON | OFF ) [-XMLMinFileSize <positive_integer>]] [-XMLBlockPI ( ON
| OFF )] [-XMLBlockDTD ( ON | OFF )] [-XMLBlockExternalEntities ( ON | OFF )] [-XMLMaxEntityExpansionsCheck (
ON | OFF ) [-XMLMaxEntityExpansions <positive_integer>]] [-XMLMaxEntityExpansionDepthCheck ( ON | OFF ) [-
XMLMaxEntityExpansionDepth <positive_integer>]] [-XMLMaxNamespacesCheck ( ON | OFF ) [-
XMLMaxNamespaces <positive_integer>]] [-XMLMaxNamespaceUriLengthCheck ( ON | OFF ) [-
XMLMaxNamespaceUriLength <positive_integer>]] [-XMLSOAPArrayCheck ( ON | OFF ) [-XMLMaxSOAPArraySize
<positive_integer>] [-XMLMaxSOAPArrayRank <positive_integer>]] | (-XMLWSIURL <expression> [-
XMLWSIChecks <string>]) | (-XMLValidationURL <expression> (-XMLRequestSchema <string> | (-XMLWSDL
<string> [-XMLAdditionalSOAPHeaders ( ON | OFF )] [-XMLEndPointCheck ( ABSOLUTE | RELATIVE )]) | -
XMLValidateSOAPEnvelope ( ON | OFF ) [-XMLResponseSchema <string>] [-XMLValidateResponse ( ON | OFF
)]) | (-XMLAttachmentURL <expression> [-XMLMaxAttachmentSizeCheck ( ON | OFF ) [-XMLMaxAttachmentSize
<positive_integer>]] [-XMLAttachmentContentTypeCheck ( ON | OFF ) [-XMLAttachmentContentType
<expression>]]) | (-XMLSQLInjection <string> [-isRegex ( REGEX | NOTREGEX )]) [-location ( ELEMENT |
ATTRIBUTE )]) | (-XMLXSS <string> [-isRegex ( REGEX | NOTREGEX )]) [-location ( ELEMENT | ATTRIBUTE )]) | -
contentType <expression> | -excludeResContentType <expression>) [-comment <string>] [-state ( ENABLED |
DISABLED )]
```

Arguments

name

Name of the profile to which to bind an exemption or rule.

startURL

Add the specified URL to the start URL list.

Enclose URLs in double quotes to ensure preservation of any embedded spaces or non-alphanumeric characters.

denyURL

Add the specified URL to the deny URL list.

Enclose URLs in double quotes to ensure preservation of any embedded spaces or non-alphanumeric characters.

fieldConsistency

Exempt the specified web form field and form action URL from the form field consistency check, or exempt the specified cookie from the cookie consistency check.

A form field consistency exemption (relaxation) consists of the following items:

* Web form field name. Name of the form field to exempt from this check.

* Form action URL. Action URL for the web form.

* IsRegex flag. The IsRegex flag, followed by YES if the form action URL is a regular expression, or NO if it is a literal string.

formActionURL

Form action URL.

isRegex

Is a regular expression?

Possible values: REGEX, NOTREGEX

cookieConsistency

A cookie consistency exemption (relaxation) consists of the following items:

* Cookie name. Name of the cookie to exempt from this check.

* IsRegex flag. The IsRegex flag, followed by YES if the cookie name is a regular expression, or NO if it is a literal string.

SQLInjection

Exempt the specified HTTP header, web form field and the form action URL, or cookie from the SQL injection check.

An SQL injection exemption (relaxation) consists of the following items:

*Item name. Name of the web form field, cookie, or HTTP header to exempt from this check.

* Form action URL. If the item to be exempted is a web form field, the action URL for the web form.

* IsRegex flag. The IsRegex flag, followed by YES if the name or form action URL is a regular expression, or NO if it is a literal string.

* Location. Location that should be examined by the SQL injection check, either FORMFIELD for web form field, HEADER for HTTP header, or COOKIE for cookie.

location

Location of XSS injection exception - XML Element or Attribute. Default location is 'ELEMENT'

Possible values: ELEMENT, ATTRIBUTE

Default value: AS_XMLLOCATION_ELEMENT

CSRFtag

Exempt the specified form field and web form from the cross-site request forgery (CSRF tagging) check.

A CSRF tagging exemption (relaxation) consists of the following items:

* Web form field name. Regular expression that describes the web form field to exempt from this check.

* Form action URL. The action URL for the web form.

CSRFFormActionURL

CSRF form action URL.

crossSiteScripting

Exempt the specified string, found in the specified HTTP header, cookie, or web form, from the cross-site scripting check.

A cross-site scripting check exemption (relaxation) consists of the following items:

* HTML to exempt. The string to exempt from the cross-site scripting check.

* URL. The URL to exempt.

* IsRegex flag. The IsRegex flag, followed by YES if the URL is a regular expression, or NO if it is a literal string.

* location. Location which should be examined by the cross-site scripting check, either FORMFIELD for web form field, HEADER for HTTP header, or COOKIE for cookie.

fieldFormat

Impose the specified format on content returned by users in the specified web form field.

A field format rule consists of the following items:

* Form field name. The name of the form field.

* Form action URL. The form action URL for the web form.

* Field type. The field type (format) to enforce on the specified web form field.

* Field format minimum length. The minimum length allowed for data in the specified field. If 0, field can be left blank.

* Field format maximum length. The maximum length allowed for data in the specified field.

* IsRegex flag. The IsRegex flag, followed by YES if the URL is a regular expression, or NO if it is a literal string.

fieldType

Field type.

fieldFormatMinLength

Field format minimum length.

Default value: 0

Minimum value: 0

Maximum value: 65535

fieldFormatMaxLength

Field format maximum length.

Default value: 65535

Minimum value: 1

Maximum value: 65535

safeObject

Protect web sites from exposing sensitive private information such as social security numbers, credit card numbers, driver's license numbers, passport numbers, and any other type of private information that can be described by a regular expression.

A safe object consists of the following items:

* Name. A name that describes the type of information that the safe object is to protect.

* Expression. PCRE-format regular expression that describes the information to be protected.

* Maximum match length. Maximum length of a matched string.

* Action. "X-Out" to mask blocked information with the letter X, or "Remove" to remove the information.

expression

Safe Object regular expression.

maxMatchLength

Maximum match length for a Safe Object expression.

Default value: 1

Minimum value: 1

Maximum value: 65535

action

Safe Object action types. (BLOCK | LEARN | LOG | STATS | NONE)

trustedLearningClients

Trusted host/network learning IP.

This binding is applicable to profile Type: HTML, XML.

comment

Any comments about the purpose of profile, or other useful information about the profile.

state

Enabled.

Possible values: ENABLED, DISABLED

Default value: ENABLED

XMLDoSURL

Exempt the specified URL from the specified XML denial-of-service (XDoS) attack protections.

An XDoS exemption (relaxation) consists of the following items:

- * URL. PCRE-format regular expression for the URL or URLs to be exempted.
- * Maximum-element-depth-check toggle. ON to enable this check, OFF to disable it.
- * Maximum-element-depth-check toggle. ON to enable, OFF to disable.
- * Maximum-element-depth-check level. Positive integer representing the maximum allowed depth of nested XML elements.
- * Maximum-element-name-length-check toggle. ON to enable, OFF to disable.
- * Maximum element name length. Positive integer representing the maximum allowed length of XML element names.
- * Maximum-number-of-elements-check toggle. ON to enable, OFF to disable.
- * Maximum number of elements. Positive integer representing the maximum allowed number of XML elements.
- * Maximum-number-of-element-children-check toggle. ON to enable, OFF to disable.
- * Maximum number of element children. Positive integer representing the maximum allowed number of XML element children.
- * Maximum-number-of-attributes-check toggle. ON to enable, OFF to disable.
- * Maximum number of attributes. Positive integer representing the maximum allowed number of XML attributes.
- * Maximum-attribute-name-length-check toggle. ON to enable, OFF to disable.
- * Maximum attribute name length. Positive integer representing the maximum allowed length of XML attribute names.

- * Maximum-attribute-value-length-check toggle. ON to enable, OFF to disable.
- * Maximum attribute value length. Positive integer representing the maximum allowed length of XML attribute values.
- * Maximum-character-data-length-check toggle. ON to enable, OFF to disable.
- * Maximum character-data length. Positive integer representing the maximum allowed length of XML character data.
- * Maximum-file-size-check toggle. ON to enable, OFF to disable.
- * Maximum file size. Positive integer representing the maximum allowed size, in bytes, of attached or uploaded files.
- * Minimum-file-size-check toggle. ON to enable, OFF to disable.
- * Minimum file size. Positive integer representing the minimum allowed size, in bytes, of attached or uploaded files.
- * Maximum-number-of-entity-expansions-check toggle. ON to enable, OFF to disable.
- * Maximum number of entity expansions. Positive integer representing the maximum allowed number of XML entity expansions.
- * Maximum-number-of XML-namespaces-check toggle. ON to enable, OFF to disable.
- * Maximum number of XML namespaces. Positive integer representing the maximum allowed number of XML namespaces.
- * Maximum-XML-namespace-URI-length-check toggle. ON to enable, OFF to disable.
- * MaximumXML-namespace URI length. Positive integer representing the maximum allowed length of XML namespace URIs.
- * Block-processing-instructions toggle. Block XML processing instructions. ON to enable, OFF to disable.
- * Block-DTD toggle. Block design type documents (DTDs). ON to enable, OFF to disable.
- * Block-external-XML-entitites toggle. ON to enable, OFF to disable.
- * Maximum-SOAP-array-check toggle. ON to enable, OFF to disable.
- * Maximum SOAP-array size. Positive integer representing the maximum allowed size of XML SOAP arrays.
- * Maximum SOAP-array rank. Positive integer representing the maximum rank (dimensions) of any single XML SOAP array.

XMLMaxElementDepthCheck

State if XML Max Element Depth Check is ON or OFF.

Possible values: ON, OFF

Default value: OFF

XMLMaxElementDepth

Maximum nesting (depth) of XML elements. This check protects against documents that have excessive depth of hierarchy.

Default value: 256

Minimum value: 1

Maximum value: 65535

XMLMaxElementNameLengthCheck

State if XML Max Element Name Length Check is ON or OFF.

Possible values: ON, OFF

Default value: OFF

XMLMaxElementNameLength

Specify the longest name of any element (including the prefix for qualified element name) to protect against overflow attacks.

Default value: 128

Minimum value: 1

Maximum value: 65535

XMLMaxElementsCheck

State if XML Max Elements Check is ON or OFF.

Possible values: ON, OFF

Default value: OFF

XMLMaxElements

Specifying maximum number of elements protects against overflow attacks.

Default value: 65535

Minimum value: 1

Maximum value: 65535

XMLMaxElementChildrenCheck

State if XML Max Element Children Check is ON or OFF.

Possible values: ON, OFF

Default value: OFF

XMLMaxElementChildren

Specifying maximum number of children allowed per element protects against overflow attacks.

Default value: 65535

Minimum value: 0

Maximum value: 65535

XMLMaxAttributesCheck

State if XML Max Attributes Check is ON or OFF.

Possible values: ON, OFF

Default value: OFF

XMLMaxAttributes

Specify maximum number of attributes per element. Protects against overflow attacks.

Default value: 256

Minimum value: 0

Maximum value: 65535

XMLMaxAttributeNameLengthCheck

State if XML Max Attribute Name Length Check is ON or OFF.

Possible values: ON, OFF

Default value: OFF

XMLMaxAttributeNameLength

Specify the longest name of any attribute (including the prefix for qualified attribute name). Protects against overflow attacks.

Default value: 128

Minimum value: 1

Maximum value: 65535

XMLMaxAttributeValueLengthCheck

State if XML Max Attribute Value Length is ON or OFF.

Possible values: ON, OFF

Default value: OFF

XMLMaxAttributeValueLength

Specify the longest value of any attribute. Protects against overflow attacks.

Default value: 2048

Minimum value: 0

Maximum value: 65535

XMLMaxCharDATALengthCheck

State if XML Max CDATA Length Check is ON or OFF.

Possible values: ON, OFF

Default value: OFF

XMLMaxCharDATALength

Maximum size of CDATA protects against overflow attacks and large unparsed data within XML messages.

Default value: 65535

Minimum value: 0

Maximum value: 1000000000

XMLMaxFileSizeCheck

State if XML Max File Size Check is ON or OFF.

Possible values: ON, OFF

Default value: OFF

XMLMaxFileSize

Maximum size of the XML messages protects against overflow attacks.

Default value: 20000000

Minimum value: 4

Maximum value: 1000000000

XMLMinFileSizeCheck

State if XML Min File Size Check is ON or OFF.

Possible values: ON, OFF

Default value: OFF

XMLMinFileSize

Enforces minimum message size.

Default value: 9

Minimum value: 4

Maximum value: 1000000000

XMLBlockPI

State if XML Block PI is ON or OFF. Protects resources from denial of service attacks as SOAP messages can not have Processing Instruction (PI) in the message.

Possible values: ON, OFF

Default value: OFF

XMLBlockDTD

State if XML DTD is ON or OFF. Protects against recursive Document Type Declaration (DTD) entity expansion attacks. Also, SOAP messages can not have DTD in the message.

Possible values: ON, OFF

Default value: OFF

XMLBlockExternalEntities

State if XML Block External Entities Check is ON or OFF. Protects against XML External Entity (XXE) attacks that force applications to parse untrusted external entities (sources) in XML documents.

Possible values: ON, OFF

Default value: OFF

XMLMaxEntityExpansionsCheck

State if XML Max Entity Expansions Check is ON or OFF.

Possible values: ON, OFF

Default value: OFF

XMLMaxEntityExpansions

Specify maximum allowed number of entity expansions. Protects against Entity Expansion Attack.

Default value: 512

Minimum value: 0

Maximum value: 1024

XMLMaxEntityExpansionDepthCheck

State if XML Max Entity Expansions Depth Check is ON or OFF.

Possible values: ON, OFF

Default value: OFF

XMLMaxEntityExpansionDepth

Specify maximum entity expansion depth. Protects against Entity Expansion Attack.

Default value: 8

Minimum value: 0

Maximum value: 24

XMLMaxNamespacesCheck

State if XML Max Namespaces Check is ON or OFF.

Possible values: ON, OFF

Default value: OFF

XMLMaxNamespaces

Specify maximum number of active namespaces. Protects against overflow attacks.

Default value: 16

Minimum value: 0

Maximum value: 512

XMLMaxNamespaceUriLengthCheck

State if XML Max Namespace URI Length Check is ON or OFF.

Possible values: ON, OFF

Default value: OFF

XMLMaxNamespaceUriLength

Specify the longest URI of any XML namespace. Protects against overflow attacks.

Default value: 256

Minimum value: 0

Maximum value: 65535

XMLSOAPArrayCheck

State if XML SOAP Array check is ON or OFF.

Possible values: ON, OFF

Default value: OFF

XMLMaxSOAPArraySize

XML Max Total SOAP Array Size. Protects against SOAP Array Abuse attack.

Default value: 20000000

Minimum value: 0

Maximum value: 1000000000

XMLMaxSOAPArrayRank

XML Max Total SOAP Array Rank. Protects against SOAP Array Abuse attack.

Default value: 16

Minimum value: 0

Maximum value: 32

XMLWSIURL

Exempt the specified URL from the web services interoperability (WS-I) check. The URL is specified as a PCRE-format regular expression, which can match one or more URLs.

XMLWSIChecks

Synonym for XMLWISURL, but takes a literal URL instead of a PCRE-format regular expression.

XMLValidationURL

Exempt the specified URL from the XML message validation check.

An XML message validation exemption (relaxation) consists of the following items:

- * URL. PCRE-format regular expression that matches the URL(s) to be exempted.
- * XML-request-schema toggle. Use the specified XML schema to validate requests. ON to enable, OFF to disable.
- * XML request schema. XML schema to use for validating requests.
- * XML-response-schema toggle. Use the specified XML schema to validate responses. ON to enable, OFF to disable.
- * XML response schema. XML schema to use for validating responses.
- * WSDL toggle. Use the specified WSDL to validate. ON to enable, OFF to disable.
- * WSDL. WSDL to use for validation.
- * SOAP-envelope toggle. Validate against the SOAP envelope. ON to enable, OFF to disable.
- * Additional-SOAP-headers toggle. Validate against the extended list of SOAP headers. ON to enable, OFF to disable.
- * XML-end-point check. ABSOLUTE to use an absolute end point, RELATIVE to use a relative end point.

XMLRequestSchema

XML Schema object for request validation .

XMLResponseSchema

XML Schema object for response validation .

XMLWSDL

WSDL object for soap request validation .

XMLAdditionalSOAPHeaders

Allow additional soap headers.

Possible values: ON, OFF

XMLEndPointCheck

Modifies the behaviour of the Request URL validation w.r.t. the Service URL.

If set to ABSOLUTE, the entire request URL is validated with the entire URL mentioned in Service of the associated WSDL.

eg: Service URL: <http://example.org/ExampleService>, Request URL: <http://example.com/ExampleService> would FAIL the validation.

If set to RELATIVE, only the non-hostname part of the request URL is validated against the non-hostname part of the Service URL.

eg: Service URL: http://example.org/ExampleService, Request URL: http://example.com/ExampleService would PASS the validation.

Possible values: ABSOLUTE, RELATIVE

Default value: ABSOLUTE

XMLValidateSOAPEnvelope

Validate SOAP Envelope only.

Possible values: ON, OFF

XMLValidateResponse

Validate response message.

Possible values: ON, OFF

XMLAttachmentURL

Exempt the specified URL from the XML attachment check.

An XML attachment exemption (relaxation) consists of the following items:

- * URL. PCRE-format regular expression that matches the URL(s) to be exempted.
- * Maximum-attachment-size-check toggle. ON to enable, OFF to disable.
- * Maximum attachment size. Positive integer representing the maximum allowed size in bytes for each XML attachment.
- * Attachment-content-type-check toggle. ON to enable, OFF to disable.
- * Attachment content type. PCRE-format regular expression that specifies the list of MIME content types allowed for XML attachments.

XMLMaxAttachmentSizeCheck

State if XML max attachment size check is ON or OFF. Protects against XML requests with large attachment data.

Possible values: ON, OFF

Default value: OFF

XMLMaxAttachmentSize

Specify maximum attachment size.

Minimum value: 0

Maximum value: 1000000000

XMLAttachmentContentTypeCheck

State if XML attachment content-type check is ON or OFF. Protects against XML requests with illegal attachments.

Possible values: ON, OFF

Default value: OFF

XMLAttachmentContentType

Specify content-type regular expression.

XMLSQLInjection

Exempt the specified URL from the XML SQL injection check.

An XML attachment exemption (relaxation) consists of the following items:

- * URL. URL to exempt, as a string or a PCRE-format regular expression.
- * ISREGEX flag. REGEX if URL is a regular expression, NOTREGEX if URL is a fixed string.
- * Location. ELEMENT if the attachment is located in an XML element, ATTRIBUTE if located in an XML attribute.

XMLXSS

Exempt the specified URL from the XML cross-site scripting (XSS) check.

An XML cross-site scripting exemption (relaxation) consists of the following items:

- * URL. URL to exempt, as a string or a PCRE-format regular expression.
- * ISREGEX flag. REGEX if URL is a regular expression, NOTREGEX if URL is a fixed string.
- * Location. ELEMENT if the attachment is located in an XML element, ATTRIBUTE if located in an XML attribute.

contentType

Add the specified content-type to the content-type list. Enclose content-type in double quotes to ensure preservation of any embedded spaces or non-alphanumeric characters.

excludeResContentType

Add the specified content-type to the response content-type list that are to be excluded from inspection. Enclose content-type in double quotes to ensure preservation

of any embedded spaces or non-alphanumeric characters.

unbind appfw profile

Unbinds the specified exemption (relaxation) or rule from the specified application firewall profile. See the bind appfw profile command for a description of the parameters.

Synopsys

```
unbind appfw profile <name> (-startURL <expression> | -denyURL <expression> | (-fieldConsistency <string> <formActionURL>) | -cookieConsistency <string> | (-SQLInjection <string> <formActionURL> [-location <location>]) | (-CSRFtag <string> <CSRFFormActionURL>) | (-crossSiteScripting <string> <formActionURL> [-location <location>]) | (-fieldFormat <string> <formActionURL>) | -safeObject <string> | -trustedLearningClients <ip_addr [/prefix][ipv6_addr[/prefix]]*> | -XMLDoSURL <expression> | -XMLWSIURL <expression> | -XMLValidationURL <expression> | -XMLAttachmentURL <expression> | (-XMLSQLInjection <string> [-location ( ELEMENT | ATTRIBUTE )]) | (-XMLXSS <string> [-location ( ELEMENT | ATTRIBUTE )]) | -contentType <expression> | -excludeResContentType <expression>)
```

Arguments

name

Name of the exemption (relaxation) or rule that you want to unbind.

startURL

Start URL regular expression.

denyURL

Deny URL regular expression.

fieldConsistency

Form field name.

formActionURL

Form action URL.

cookieConsistency

Cookie name.

SQLInjection

Form field, header or cookie name.

location

Location of XSS injection exception - XML Element or Attribute. Default location is 'ELEMENT'

Possible values: ELEMENT, ATTRIBUTE

Default value: AS_XMLLOCATION_ELEMENT

CSRFtag

CSRF Form origin URL.

This binding is applicable to Profile Type: HTML.

CSRFFormActionURL

CSRF form action URL.

crossSiteScripting

Form field, header or cookie name.

fieldFormat

Field format name.

safeObject

Safe Object name.

trustedLearningClients

Trusted learning Clients IP

XMLDoSURL

XML DoS URL regular expression.

XMLWSIURL

XML WS-I URL regular expression.

XMLValidationURL

XML Message URL regular expression.

XMLAttachmentURL

XML Attachment URL regular expression.

XMLSQLInjection

Exempt the specified URL from the XML SQL injection check.

An XML attachment exemption (relaxation) consists of the following items:

- * URL. URL to exempt, as a string or a PCRE-format regular expression.
- * ISREGEX flag. REGEX if URL is a regular expression, NOTREGEX if URL is a fixed string.

* Location. ELEMENT if the attachment is located in an XML element, ATTRIBUTE if located in an XML attribute.

XMLXSS

Exempt the specified URL from the XML cross-site scripting (XSS) check.

An XML cross-site scripting exemption (relaxation) consists of the following items:

* URL. URL to exempt, as a string or a PCRE-format regular expression.

* ISREGEX flag. REGEX if URL is a regular expression, NOTREGEX if URL is a fixed string.

* Location. ELEMENT if the attachment is located in an XML element, ATTRIBUTE if located in an XML attribute.

contentType

content-type regular expression.

excludeResContentType

Response content type regular expression that are to be excluded from inspection.

show appfw profile

Displays details of the specified application firewall profile. If no profile is specified, displays a list of all application firewall profiles on the NetScaler appliance.

Synopsys

show appfw profile [<name>]

Arguments

name

Name of the application firewall profile.

Outputs

stateflag

type

The profile type of of this Application Firewall profile. If the profile is of the HTML type, only checks relevant to HTML are applied. If the profile is of the XML type, only checks relevent to XML are applied. if the profile is of the Web 2.0 type, then both types of checks are applied.

defaults

Default configuration to apply to the profile. Basic defaults are intended for standard content that requires little further configuration, such as static web site content. Advanced defaults are intended for specialized content that requires significant specialized configuration, such as heavily scripted or dynamic content.

CLI users: When adding an application firewall profile, you can set either the defaults or the type, but not both. To set both options, create the profile by using the add appfw profile command, and then use the set appfw profile command to configure the other option.

useHTMLErrorObject

Send an imported HTML Error object to a user when a request is blocked, instead of redirecting the user to the designated Error URL.

errorURL

The error page for this profile.

HTMLErrorObject

Name to assign to the HTML Error Object.

Must begin with a letter, number, or the underscore character `\\(_\\)`, and must contain only letters, numbers, and the hyphen `\\(-\\)`, period `\\(.\\)` pound `\\(\\#\\)`, space `\\(\\)`, at `\\(@\\)`, equals `\\(=\\)`, colon `\\(:\\)`, and underscore characters. Cannot be changed after the HTML error object is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks `\\(for example, "my HTML error object" or 'my HTML error object'\\)`.

logEveryPolicyHit

Log every profile match, regardless of security checks results.

stripComments

Tells the Application Firewall to strip HTML comments from responses before sending them to the user.

stripHtmlComments

Tells the Application Firewall to strip HTML comments from responses before sending them to the user.

stripXmlComments

Tells the Application Firewall to strip XML comments from responses before sending them to the user.

defaultCharSet

The default character set. The character set that the Application Firewall uses for web pages that do not explicitly set a different character set.

postBodyLimit

The maximum body size for an HTTP POST.

fileUploadMaxNum

Maximum allowed number of file uploads per form-submission request. The maximum setting (65535) allows an unlimited number of uploads.

canonicalizeHTMLResponse

Tells the Application Firewall to convert any non-ASCII characters into HTML entities before sending responses to the user. This is called 'canonicalization' of HTML responses.

enableFormTagging

Enables tagging of web forms for form field Consistency checks.

sessionlessFieldConsistency

Enable session less form field consistency checks.

sessionlessURLClosure

Enable session less URL closure checks.

semicolonFieldSeparator

Allow ';' as a form field separator in URL queries and POST form bodies.

excludeFileUploadFromChecks

Excludes uploaded files from all web form checks.

SQLInjectionParseComments

Canonicalizes SQL Comments in form fields.

checkRequestHeaders

Check request headers as well as web forms for injected SQL and cross-site scripts.

optimizePartialReqs

Optimize handle of HTTP partial requests i.e. those with range headers.

Available settings are as follows:

- * ON - Partial requests by the client result in partial requests to the backend server in most cases.
- * OFF - Partial requests by the client are changed to full requests to the backend server

URLDecodeRequestCookies

URL Decode request cookies before subjecting them to SQL and cross-site scripting checks.

comment

Comments associated with this profile.

startURLAction

Start URL action types. (BLOCK | LEARN | LOG | STATS | NONE)

contentTypeAction

Content-type action types. (BLOCK | LOG | NONE)

startURL

A regular expression that designates a URL on the Start URL list.

startURLClosure

Enable Start URL closure. When enabled, this feature allows users to start their session at a designated start URL, then navigate from that start URL to any URL on a protected web site by clicking a link on another web page on that web site. Otherwise, requests to any URL that is not explicitly allowed are blocked.

denyURLAction

Deny URL action types. (BLOCK | LOG | STATS | NONE)

denyURL

A regular expression that designates a URL on the Deny URL list.

RefererHeaderCheck

Enable validation of Referer headers.

Referer validation ensures that a web form that a user sends to your web site originally came from your web site, not an outside attacker.

Although this parameter is part of the Start URL check, referer validation protects against cross-site request forgery (CSRF) attacks, not Start URL attacks.

CSRFTagAction

Cross-site request forgery tagging action types. (BLOCK | LEARN | LOG | STATS | NONE)

CSRFTag

The web form originating URL.

CSRFFormActionURL

The web form action URL.

crossSiteScriptingAction

Cross-site scripting action types. (BLOCK | LEARN | LOG | STATS | NONE)

crossSiteScriptingTransformUnsafeHTML

Enables transformation of unsafe HTML into safe HTML before forwarding a request to the web server.

crossSiteScriptingCheckCompleteURLs

Tells the Application Firewall to check complete URLs rather than just the query portion of URLs for cross-site scripting violations.

crossSiteScripting

The web form field name.

isRegex

Is the XML XSS exempted field name a regular expression?

formActionURL

Action URL of the form field to which a field format will be assigned.

exemptClosureURLsFromSecurityChecks

Tells the Application Firewall to exempt closure URLs from security checks.

location

Location of XSS injection exception - XML Element or Attribute.

SQLInjectionAction

SQL injection action types. (BLOCK | LEARN | LOG | STATS | NONE)

SQLInjectionTransformSpecialChars

Enables transformation of SQL special characters found in web forms into safe equivalents.

SQLInjectionOnlyCheckFieldsWithSQLChars

Tells the Application Firewall to check form fields that contain SQL special characters only, rather than all form fields, for SQL injection violations.

SQLInjectionType

Available SQL Injection types.

SQLInjectionCheckSQLWildChars

Check for form fields that contain SQL wild chars .

SQLInjection

The web form field name.

invalidPercentHandling

Configure the method that the application firewall uses to handle percent-encoded names and values. Available settings function as follows:

- * apache_mode - Apache format.
- * asp_mode - Microsoft ASP format.
- * secure_mode - Secure format.

fieldConsistencyAction

Form Field Consistency action types. (BLOCK | LEARN | LOG | STATS | NONE)

fieldConsistency

The web form field name.

cookieConsistencyAction

Cookie consistency action types. (BLOCK | LEARN | LOG | STATS | NONE)

cookieConsistency

The name of the cookie to be checked.

cookieTransforms

Perform the specified type of cookie transformation.

Available settings function as follows:

- * Encryption - Encrypt cookies.
- * Proxying - Mask contents of server cookies by sending proxy cookie to users.
- * Cookie flags - Flag cookies as HTTP only to prevent scripts on user's browser from accessing and possibly modifying them.

CAUTION: Make sure that this parameter is set to ON if you are configuring any cookie transformations. If it is set to OFF, no cookie transformations are performed regardless of any other settings.

cookieEncryption

Type of cookie encryption. Available settings function as follows:

- * None - Do not encrypt cookies.
- * Decrypt Only - Decrypt encrypted cookies, but do not encrypt cookies.
- * Encrypt Session Only - Encrypt session cookies, but not permanent cookies.
- * Encrypt All - Encrypt all cookies.

cookieProxying

Proxies server cookies using the Application Firewall session

addCookieFlags

Add the specified flags to cookies. Available settings function as follows:

- * None - Do not add flags to cookies.
- * HTTP Only - Add the HTTP Only flag to cookies, which prevents scripts from accessing cookies.
- * Secure - Add Secure flag to cookies.
- * All - Add both HTTPOnly and Secure flags to cookies.

bufferOverflowAction

Buffer overflow action types. (BLOCK | LOG | STATS | NONE)

bufferOverflowMaxURLLength

Maximum allowed length for URLs.

bufferOverflowMaxHeaderLength

Maximum allowed length for HTTP headers.

bufferOverflowMaxCookieLength

Maximum allowed length for cookies.

fieldFormatAction

Field format action types. (BLOCK | LEARN | LOG | STATS | NONE)

defaultFieldFormatType

Name of the default field type, the field type that the Application Firewall will assign to a form field when no specific field type is assigned to that particular form field.

defaultFieldFormatMinLength

Default field type minimum length setting.

defaultFieldFormatMaxLength

Default field type maximum length setting.

fieldFormat

Name of the form field to which a field format will be assigned.

fieldType

The field type you are assigning to this form field.

fieldFormatMinLength

The minimum allowed length for data in this form field.

fieldFormatMaxLength

The maximum allowed length for data in this form field.

creditCardAction

Credit Card action types. (BLOCK | LOG | STATS | NONE)

creditCard

Credit card types. (AMEX | DINERSCLUB | DISCOVER | JBC | MASTERCARD | VISA)

creditCardMaxAllowed

Maximum number of times a credit card number may be seen before action is taken.

creditCardXOut

X-out credit card numbers.

safeObject

Name of the Safe Object.

expression

A regular expression that defines the Safe Object.

maxMatchLength

Maximum match length for a Safe Object expression.

action

Safe Object action types. (BLOCK | LOG | STATS | NONE)

requestContentType

Default content-type for request messages.

responseContentType

Default content-type for response messages.

XMLErrorObject

URL for the xml error page

signatures

Signatures for the profile

XMLFormatAction

XML well-formed request action types. (BLOCK | LOG | STATS | NONE)

XMLDoSAction

XML DOS action types. (BLOCK | LEARN | LOG | STATS | NONE)

XMLSQLInjectionAction

XML SQL Injection action types. (BLOCK | LOG | STATS | NONE)

XMLSQLInjectionOnlyCheckFieldsWithSQLChars

XML flag to check only fields with SQL characters.

XMLSQLInjectionType

Available XML SQL Injection types.

XMLSQLInjectionCheckSQLWildChars

XML flag to check for SQL wild chars.

XMLSQLInjectionParseComments

Canonicalize SQL Comments in XML data.

XMLXSSAction

XML cross-site scripting action types. (BLOCK | LOG | STATS | NONE)

XMLWSIAction

XML WSI action types. (BLOCK | LEARN | LOG | STATS | NONE)

XMLAttachmentAction

XML attachment action types. (BLOCK | LEARN | LOG | STATS | NONE)

XMLValidationAction

XML message validation action types. (BLOCK | LOG | STATS | NONE)

XMLSOAPFaultAction

XML SOAP fault filtering action types. (BLOCK | LOG | STATS | REMOVE | NONE)

XMLDoSURL

XML DoS URL regular expression length.

XMLWSIURL

XML WS-I URL regular expression length.

XMLValidationURL

XML Validation URL regular expression.

XMLAttachmentURL

XML attachment URL regular expression length.

XMLSQLInjection

Exempt the specified URL from the XML SQL injection check.

An XML attachment exemption (relaxation) consists of the following items:

- * URL. URL to exempt, as a string or a PCRE-format regular expression.
- * ISREGEX flag. REGEX if URL is a regular expression, NOTREGEX if URL is a fixed string.
- * Location. ELEMENT if the attachment is located in an XML element, ATTRIBUTE if located in an XML attribute.

XMLXSS

Exempt the specified URL from the XML cross-site scripting (XSS) check.

An XML cross-site scripting exemption (relaxation) consists of the following items:

- * URL. URL to exempt, as a string or a PCRE-format regular expression.
- * ISREGEX flag. REGEX if URL is a regular expression, NOTREGEX if URL is a fixed string.
- * Location. ELEMENT if the attachment is located in an XML element, ATTRIBUTE if located in an XML attribute.

state

Enabled.

XMLMaxElementDepthCheck

State if XML Max element depth check is ON or OFF.

XMLMaxElementDepth

Maximum nesting (depth) of XML elements. This check protects against documents that have excessive hierarchy depths.

XMLMaxElementNameLengthCheck

State if XML Max element name length check is ON or OFF.

XMLMaxElementNameLength

Specify the longest name of any element (including the expanded namespace) to protect against overflow attacks.

XMLMaxElementsCheck

State if XML Max elements check is ON or OFF.

XMLMaxElements

Specify the maximum number of XML elements allowed. Protects against overflow attacks.

XMLMaxElementChildrenCheck

State if XML Max element children check is ON or OFF.

XMLMaxElementChildren

Specify the maximum number of children allowed per XML element. Protects against overflow attacks.

XMLMaxNodesCheck

State if XML Max nodes check is ON or OFF.

XMLMaxNodes

Specify the maximum number of XML nodes. Protects against overflow attacks.

XMLMaxAttributesCheck

State if XML Max attributes check is ON or OFF.

XMLMaxAttributes

Specify maximum number of attributes per XML element. Protects against overflow attacks.

XMLMaxAttributeNameLengthCheck

State if XML Max attribute name length check is ON or OFF.

XMLMaxAttributeNameLength

Specify the longest name of any XML attribute. Protects against overflow attacks.

XMLMaxAttributeValueLengthCheck

State if XML Max attribute value length is ON or OFF.

XMLMaxAttributeValueLength

Specify the longest value of any XML attribute. Protects against overflow attacks.

XMLMaxCharDATALengthCheck

State if XML Max CDATA length check is ON or OFF.

XMLMaxCharDATALength

Specify the maximum size of CDATA. Protects against overflow attacks and large quantities of unparsed data within XML messages.

XMLMaxFileSizeCheck

State if XML Max file size check is ON or OFF.

XMLMaxFileSize

Specify the maximum size of XML messages. Protects against overflow attacks.

XMLMinFileSizeCheck

State if XML Min file size check is ON or OFF.

XMLMinFileSize

Enforces minimum message size.

XMLBlockPI

State if XML Block PI is ON or OFF. Protects resources from denial of service attacks as SOAP messages cannot have processing instructions (PI) in messages.

XMLBlockDTD

State if XML DTD is ON or OFF. Protects against recursive Document Type Declaration (DTD) entity expansion attacks. Also, SOAP messages cannot have DTDs in messages.

XMLBlockExternalEntities

State if XML Block External Entities Check is ON or OFF. Protects against XML External Entity (XXE) attacks that force applications to parse untrusted external entities (sources) in XML documents.

XMLMaxEntityExpansionsCheck

State if XML Max Entity Expansions Check is ON or OFF.

XMLMaxEntityExpansions

Specify maximum allowed number of entity expansions. Protects against Entity Expansion Attack.

XMLMaxEntityExpansionDepthCheck

State if XML Max Entity Expansions Depth Check is ON or OFF.

XMLMaxEntityExpansionDepth

Specify maximum entity expansion depth. Protects against Entity Expansion Attack.

XMLMaxNamespacesCheck

State if XML Max namespaces check is ON or OFF.

XMLMaxNamespaces

Specify maximum number of active namespaces. Protects against overflow attacks.

XMLMaxNamespaceUriLengthCheck

State if XML Max namespace URI length check is ON or OFF.

XMLMaxNamespaceUriLength

Specify the longest URI of any XML namespace. Protects against overflow attacks.

XMLSOAPArrayCheck

State if XML SOAP Array check is ON or OFF.

XMLMaxSOAPArraySize

XML Max Total SOAP Array Size. Protects against SOAP Array Abuse attack.

XMLMaxSOAPArrayRank

XML Max Individual SOAP Array Rank. This is the dimension of the SOAP array.

XMLWSIChecks

Specify a comma separated list of relevant WS-I rule IDs. (R1140, R1141)

XMLRequestSchema

XML Schema object for request validation .

XMLResponseSchema

XML Schema object for response validation.

XMLWSDL

WSDL object for soap request validation.

XMLAdditionalSOAPHeaders

Allow additional soap headers.

XMLEndPointCheck

Modifies the behaviour of the Request URL validation w.r.t. the Service URL.

If set to ABSOLUTE, the entire request URL is validated with the entire URL mentioned in Service of the associated WSDL.

eg: Service URL: <http://example.org/ExampleService>, Request URL: <http://example.com/ExampleService> would FAIL the validation.

If set to RELATIVE, only the non-hostname part of the request URL is validated against the non-hostname part of the Service URL.

eg: Service URL: http://example.org/ExampleService, Request URL: http://example.com/ExampleService would PASS the validation.

XMLValidateSOAPEnvelope

Validate SOAP Envelope only.

XMLValidateResponse

Validate response message.

XMLMaxAttachmentSizeCheck

State if XML Max attachment size Check is ON or OFF. Protects against XML requests with large attachment data.

XMLMaxAttachmentSize

Specify maximum attachment size.

XMLAttachmentContentTypeCheck

State if XML attachment content-type check is ON or OFF. Protects against XML requests with illegal attachments.

XMLAttachmentContentType

Specify content-type regular expression.

builtin

Indicates that a profile is a built-in entity.

builtinType

Type of built-in profiles

trustedLearningClients

Specify trusted host/network IP

contentType

A regular expression that designates a content-type on the content-types list.

excludeResContentType

A regular expression that represents the content type of the response that are to be excluded from inspection.

devno

count

stat appfw profile

Displays statistics for the specified application firewall profile. If no profile is specified, displays abbreviated statistics for all profiles.

Synopsys

stat appfw profile [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

name

Name of the application firewall profile.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count**devno****stateflag**

Outputs

requests (reqs)

HTTP/HTTPS requests sent to your protected web servers via the Application Firewall.

Request Bytes (reqBytes)

Number of bytes transferred for requests

responses (resps)

HTTP/HTTPS responses sent by your protected web servers via the Application Firewall.

Response Bytes (resBytes)

Number of bytes transferred for responses

aborts

Incomplete HTTP/HTTPS requests aborted by the client before the Application Firewall could finish processing them.

redirects (redirect)

HTTP/HTTPS requests redirected by the Application Firewall to a different Web page or web server. (HTTP 302)

Long Term Ave Response Time (ms) (longAvgRespTimePP)

Average backend response time in milliseconds since reboot

Recent Ave Response Time (ms) (shortAvgRespTimePP)

Average backend response time in milliseconds over the last 7 seconds

start URL (startURL)

Number of Start URL security check violations seen by the Application Firewall.

deny URL (denyURL)

Number of Deny URL security check violations seen by the Application Firewall.

referer header (refererHdr)

Number of Referer Header security check violations seen by the Application Firewall.

buffer overflow (bufovfl)

Number of Buffer Overflow security check violations seen by the Application Firewall.

cookie consistency (cookie)

Number of Cookie Consistency security check violations seen by the Application Firewall.

CSRF form tag (csrf_tag)

Number of Cross Site Request Forgery form tag security check violations seen by the Application Firewall.

HTML Cross-site scripting (xss)

Number of HTML Cross-Site Scripting security check violations seen by the Application Firewall.

HTML SQL injection (sql)

Number of HTML SQL Injection security check violations seen by the Application Firewall.

field format (fieldfmt)

Number of Field Format security check violations seen by the Application Firewall.

field consistency (fieldcon)

Number of Field Consistency security check violations seen by the Application Firewall.

credit card (ccard)

Number of Credit Card security check violations seen by the Application Firewall.

safe object (safeobj)

Number of Safe Object security check violations seen by the Application Firewall.

Signature Violations (sigs)

Number of Signature violations seen by the Application Firewall.

XML Format (wfcViolations)

Number of XML Format security check violations seen by the Application Firewall.

XML Denial of Service (XDoS) (xdosViolations)

Number of XML Denial-of-Service security check violations seen by the Application Firewall.

XML Message Validation (msgvalViolations)

Number of XML Message Validation security check violations seen by the Application Firewall.

Web Services Interoperability (wsIViolations)

Number of Web Services Interoperability (WS-I) security check violations seen by the Application Firewall.

XML SQL Injection (xmlSqlViolations)

Number of XML SQL Injection security check violations seen by the Application Firewall.

XML Cross-Site Scripting (xmlXssViolations)

Number of XML Cross-Site Scripting (XSS) security check violations seen by the Application Firewall.

XML Attachment (xmlAttachmentViolations)

Number of XML Attachment security check violations seen by the Application Firewall.

SOAP Fault Violations (soapflt)

Number of requests returning soap:fault from the backend server

XML Generic Violations (genflt)

Number of requests returning XML generic violation from the backend server

Total Violations (totperpr)

Number of violations seen by the application firewall on per profile basis

HTTP Client Errors (4xx Resp) (4xxResps)

Number of requests returning HTTP 4xx from the backend server

HTTP Server Errors (5xx Resp) (5xxResps)

Number of requests returning HTTP 5xx from the backend server

Example

```
stat appfw profile
```

archive appfw profile

Create archive for the profile.

Synopsys

```
archive appfw profile <name> <archivename> [-comment <string>]
```

Arguments

name

Name for the profile. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.), pound (#), space (), at (@), equals (=), colon (:), and underscore (_) characters. Cannot be changed after the profile is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my profile" or 'my profile').

archivename

Source for tar archive.

comment

Any comments about the purpose of profile, or other useful information about the profile.

restore appfw profile

Restore configuration from archive file

Synopsys

restore appfw profile <archivename>

Arguments

archivename

Source for tar archive.

appfw settings

The following operations can be performed on "appfw settings":

[set](#) | [unset](#) | [show](#)

set appfw settings

Modifies the global application firewall settings. The global settings apply to all application firewall profiles.

Synopsys

```
set appfw settings [-defaultProfile <string>] [-undefAction <string>] [-sessionTimeout <positive_integer>] [-learnRateLimit <positive_integer>] [-sessionLifetime <positive_integer>] [-sessionCookieName <string>] [-clientIPLoggingHeader <string>] [-importSizeLimit <positive_integer>] [-signatureAutoUpdate ( ON | OFF )] [-signatureUrl <expression>] [-cookiePostEncryptPrefix <string>] [-logMalformedReq ( ON | OFF )] [-CEFLogging ( ON | OFF )] [-entityDecoding ( ON | OFF )] [-useConfigurableSecretKey ( ON | OFF )]
```

Arguments

defaultProfile

Profile to use when a connection does not match any policy. Default setting is APPFW_BYPASS, which sends unmatched connections back to the NetScaler appliance without attempting to filter them further.

Default value: APPFW_BYPASS

undefAction

Profile to use when an application firewall policy evaluates to undefined (UNDEF).

An UNDEF event indicates an internal error condition. The APPFW_BLOCK built-in profile is the default setting. You can specify a different built-in or user-created profile as the UNDEF profile.

Default value: APPFW_BLOCK

sessionTimeout

Timeout, in seconds, after which a user session is terminated. Before continuing to use the protected web site, the user must establish a new session by opening a designated start URL.

Default value: 900

Minimum value: 1

Maximum value: 65535

learnRateLimit

Maximum number of connections per second that the application firewall learning engine examines to generate new relaxations for learning-enabled security checks. The application firewall drops any connections above this limit from the list of connections used by the learning engine.

Default value: 400

Minimum value: 1

Maximum value: 1000

sessionLifetime

Maximum amount of time (in seconds) that the application firewall allows a user session to remain active, regardless of user activity. After this time, the user session is terminated. Before continuing to use the protected web site, the user must establish a new session by opening a designated start URL.

Default value: 0

Minimum value: 0

Maximum value: 2147483647

sessionCookieName

Name of the session cookie that the application firewall uses to track user sessions.

Must begin with a letter or number, and can consist of from 1 to 31 letters, numbers, and the hyphen (-) and underscore (_) symbols.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my cookie name" or 'my cookie name').

Default value: NS_S_AS_DEFAULT_COOKIE_NAME

clientIPLoggingHeader

Name of an HTTP header that contains the IP address that the client used to connect to the protected web site or service.

importSizeLimit

Cumulative total maximum number of bytes in web forms imported to a protected web site. If a user attempts to upload files with a total byte count higher than the specified limit, the application firewall blocks the request.

Default value: 134217728

Minimum value: 1

Maximum value: 134217728

signatureAutoUpdate

Flag used to enable/disable auto update signatures

Possible values: ON, OFF

Default value: OFF

signatureUri

URL to download the mapping file from server

Default value: <https://s3.amazonaws.com/NSAppFwSignatures/SignaturesMapping.xml>

cookiePostEncryptPrefix

String that is prepended to all encrypted cookie values.

Default value: NS_S_AS_DEFAULT_CKI_POST_ENCRYPT_PREFIX

logMalformedReq

Log requests that are so malformed that application firewall parsing doesn't occur.

Possible values: ON, OFF

Default value: ON

CEFLogging

Enable CEF format logs.

Possible values: ON, OFF

Default value: OFF

entityDecoding

Transform multibyte (double- or half-width) characters to single width characters.

Possible values: ON, OFF

Default value: OFF

useConfigurableSecretKey

Use configurable secret key in AppFw operations

Possible values: ON, OFF

Default value: OFF

unset appfw settings

Use this command to remove appfw settings settings. Refer to the set appfw settings command for meanings of the arguments.

Synopsis

unset appfw settings [-defaultProfile] [-undefAction] [-sessionTimeout] [-learnRateLimit] [-sessionLifetime] [-sessionCookieName] [-clientIPLoggingHeader] [-importSizeLimit] [-signatureAutoUpdate] [-signatureUrl] [-cookiePostEncryptPrefix] [-logMalformedReq] [-CEFLogging] [-entityDecoding] [-useConfigurableSecretKey]

show appfw settings

Displays the current application firewall global settings.

Synopsis

show appfw settings

Outputs

defaultProfile

Profile to use when a connection does not match any policy. Default setting is APPFW_BYPASS, which sends unmatched connections back to the NetScaler appliance without attempting to filter them further.

undefAction

Profile to use when an application firewall policy evaluates to undefined (UNDEF).

An UNDEF event indicates an internal error condition. The APPFW_BLOCK built-in profile is the default setting. You can specify a different built-in or user-created profile as the UNDEF profile.

sessionTimeout

Session timeout (in seconds).

learnRateLimit

Learn messages rate limit value (in messages per second).

sessionLifetime

Session lifetime (in seconds). Zero means no limit.

sessionCookieName

Name of the session cookie that the application firewall uses to track user sessions.

Must begin with a letter or number, and can consist of from 1 to 31 letters, numbers, and the hyphen (-) and underscore (_) symbols.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my cookie name" or 'my cookie name').

clientIPLoggingHeader

Name of header that holds downstream IP address for logging purposes.

importSizeLimit

Cumulative total maximum number of bytes in web forms imported to a protected web site. If a user attempts to upload files with a total byte count higher than the specified limit, the application firewall blocks the request.

signatureAutoUpdate

Flag used to enable/disable auto update signatures

signatureUrl

URL to download the mapping file from server

cookiePostEncryptPrefix

String that is prepended to all encrypted cookie values.

logMalformedReq

Log requests that are so malformed that application firewall parsing doesn't occur.

CEFLogging

Enable CEF format logs.

entityDecoding

Transform multibyte (double- or half-width) characters to single width characters.

useConfigurableSecretKey

Use configurable secret key in AppFw operations

appfw signatures

The following operations can be performed on "appfw signatures":

[rm](#) | [show](#) | [import](#) | [update](#)

rm appfw signatures

Removes the specified signature object from the application firewall.

Synopsis

```
rm appfw signatures <name>
```

Arguments

name

Name of the signature object.

Example

```
rm signatures <name>
```

show appfw signatures

Displays the specified signatures object. If no signatures object is specified, displays all signatures objects defined on the NetScaler appliance.

Synopsis

```
show appfw signatures [<name>]
```

Arguments

name

Name of the signature object.

Outputs

response

Example

```
show appfw signatures
```

import appfw signatures

Imports the specified signatures object to the NetScaler appliance and assigns it the specified name.

Synopsis

```
import appfw signatures <src> <name> [-xslt <string>] [-comment <string>] [-overwrite] [-merge] [-sha1 <string>]
```

Arguments

src

URL (protocol, host, path, and file name) for the location at which to store the imported signatures object.

NOTE: The import fails if the object to be imported is on an HTTPS server that requires client certificate authentication for access.

name

Name to assign to the signatures object on the NetScaler appliance.

xslt

XSLT file source.

comment

Any comments to preserve information about the signatures object.

overwrite

Overwrite any existing signatures object of the same name.

merge

Merges the existing Signature with new signature rules

sha1

File path for sha1 file to validate signature file

Example

```
import signatures http://www.example.com/ns/signatures.xml my-signature
```

update appfw signatures

Updates the specified signatures object from the source.

Synopsys

```
update appfw signatures <name> [-mergeDefault]
```

Arguments

name

Name of the signatures object to update.

mergeDefault

Merges signature file with default signature file.

Example

```
update signatures my-signatures
```

appfw stats

The following operations can be performed on "appfw stats":

show appfw stats

show appfw stats is an alias for stat appfw

Synopsys

show appfw stats - alias for 'stat appfw'

appfw transactionRecords

The following operations can be performed on "appfw transactionRecords":

show appfw transactionRecords

Display an application firewall transaction record.

Synopsys

show appfw transactionRecords

Outputs

httpTransactionId

The http transaction identifier.

packetEngineId

The packet engine identifier.

AppFwSessionId

The session identifier set by the Application Firewall to track the user session.

profileName

Application Firewall profile name.

url

Request URL

clientip

The IP address of client.

destIP

The IP address of destination.

startTime

Conveys time at which request processing started.

endTime

Conveys time at which request processing end.

requestContentLength

The content length of request.

requestYields

The number of times yielded during request processing to send heart beat packets.

requestMaxProcessingTime

The maximum processing time across yields during request processing.

responseContentLength

The content length of response.

responseYields

The number of times yielded during response processing to send heart beat packets.

responseMaxProcessingTime

The maximum processing time across yields during response processing.

flag

Record flags.

devno

count

stateflag

appfw wsdl

The following operations can be performed on "appfw wsdl":

[rm](#) | [show](#) | [import](#)

rm appfw wsdl

Removes the specified imported WSDL file from the application firewall.

Synopsys

```
rm appfw wsdl <name>
```

Arguments

name

Name of the WSDL file to remove.

Example

```
rm wsdl <name>
```

show appfw wsdl

Removes the specified imported WSDL file.

Synopsys

```
show appfw wsdl [<name>]
```

Arguments

name

Name of the WSDL file to display.

Outputs

response

Example

```
show appfw wsdl
```

import appfw wsdl

Imports the specified WSDL file to the application firewall.

Synopsys

```
import appfw wsdl <src> <name> [-comment <string>] [-overwrite]
```

Arguments

src

URL (protocol, host, path, and name) of the WSDL file to be imported is stored.

NOTE: The import fails if the object to be imported is on an HTTPS server that requires client certificate authentication for access.

name

Name to assign to the WSDL on the NetScaler appliance.

comment

Any comments to preserve information about the WSDL.

overwrite

Overwrite any existing WSDL of the same name.

Example

```
import appfw wsdl http://www.webs servicex.net/stockquote.asmx?wsdl stockquote
```

appfw xmlerrorpage

The following operations can be performed on "appfw xmlerrorpage":

[rm](#) | [show](#) | [import](#) | [update](#)

rm appfw xmlerrorpage

Removes the object imported by import xmlerrorpage.

Synopsis

```
rm appfw xmlerrorpage <name>
```

Arguments

name

Indicates name of the imported xml error page to be removed.

Example

```
rm xmlerrorpage <name>
```

show appfw xmlerrorpage

Displays the specified XML error object. If no XML error page object is specified, displays a list of all XML error objects on the NetScaler appliance.

Synopsis

```
show appfw xmlerrorpage [<name>]
```

Arguments

name

Name of the XML error object.

Outputs

response

Example

```
show appfw xmlerrorpage
```

import appfw xmlerrorpage

Imports the specified XML error page to the NetScaler appliance and assigns it the specified name.

Synopsis

```
import appfw xmlerrorpage <src> <name> [-comment <string>] [-overwrite]
```

Arguments

src

URL (protocol, host, path, and name) for the location at which to store the imported XML error object.

NOTE: The import fails if the object to be imported is on an HTTPS server that requires client certificate authentication for access.

name

Name to assign to the XML error object on the NetScaler appliance.

comment

Any comments to preserve information about the XML error object.

overwrite

Overwrite any existing XML error object of the same name.

Example

```
import xmlerrorpage http://www.example.com/errorpage.xml my-xml-error-page
```

update appfw xmlerrorpage

Updates the specified XML error object from the source.

Synopsys

```
update appfw xmlerrorpage <name>
```

Arguments

name

Name of the XML error object.

Example

```
update xmlerrorpage my-xml-error-page
```

appfw xmlschema

The following operations can be performed on "appfw xmlschema":

[rm](#) | [show](#) | [import](#)

rm appfw xmlschema

Removes the specified XML Schema object from the application firewall.

Synopsis

```
rm appfw xmlschema <name>
```

Arguments

name

Name of the XML Schema object to remove.

Example

```
rm xmlschema <name>
```

show appfw xmlschema

Displays the specified XML Schema object. If no object is specified, displays all XML Schema objects on the NetScaler appliance.

Synopsis

```
show appfw xmlschema [<name>]
```

Arguments

name

Name of the XML Schema object to display.

Outputs

response

Example

```
show appfw xmlschema
```

import appfw xmlschema

Imports the specified XML Schema to the NetScaler appliance and assigns it the specified name.

Synopsis

```
import appfw xmlschema <src> <name> [-comment <string>] [-overwrite]
```

Arguments

src

URL (protocol, host, path, and file name) for the location at which to store the imported XML Schema.

NOTE: The import fails if the object to be imported is on an HTTPS server that requires client certificate authentication for access.

name

Name to assign to the XML Schema object on the NetScaler appliance.

comment

Any comments to preserve information about the XML Schema object.

overwrite

Overwrite any existing XML Schema object of the same name.

Example

```
import xmlschema http://schemas.xmlsoap.org/soap/envelope/ soap
```

AppQoE Commands

The entities on which you can perform NetScaler CLI operations:

- appqoe
- appqoe CustomResp
- appqoe action
- appqoe parameter
- appqoe policy
- appqoe stats

appqoe

The following operations can be performed on "appqoe":

stat appqoe

Displays statistics of feature AppQoE.

Synopsys

```
stat appqoe [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

In-Memory responses sent (TotInMemRsp)

Total in-memory responses sent from NS

Faulty cookies received (TotFaultyCookies)

Total faulty cookies received

Valid cookies received (TotValidCookies)

Total valid cookies received

High priority requests served (TotHighPriReq)

Total Requests served from higher priority queue

Medium priority requests served (TotMediumPriReq)

Total Requests served from medium priority queue

Low priority requests served (TotLowPriReq)

Total Requests served from low priority queue

Lowest(Surge) priority requests served (TotLowestPriReq)

Total Requests served from surge priority queue

Alt. server substitution failed (TotAltSvrSubFailed)

Total number of times alternate server substitution failed

HDOS condition triggered (TotDoSTrig)

Total number of times HDOS condition triggered

Valid DOSQ cookies received (TotDOSQValidCookies)

Total DOSQ valid cookies received

Valid DOSH cookies received (TotDOSHValidCookies)

Total DOSH valid cookies received

Valid SID cookies received (TotSIDValidCookies)

Total SID valid cookies received

Valid ONH cookies received (TotONHValidCookies)

Total ONH valid cookies received

Valid PRIQ cookies received (TotPRIQValidCookies)

Total PRIQ valid cookies received

Faulty DOSQ cookies received (TotDOSQFaultyCookies)

Total DOSQ faulty cookies received

Faulty DOSH cookies received (TotDOSHFaultyCookies)

Total DOSH faulty cookies received

Faulty SID cookies received (TotSIDFaultyCookies)

Total SID faulty cookies received

Faulty ONH cookies received (TotONHFaultyCookies)

Total ONH faulty cookies received

Faulty PRIQ cookies received (TotPRIQFaultyCookies)

Total PRIQ faulty cookies received

Requests for valid embedded links (TotPRIEmbedLinks)

Total requests for valid embedded links

Valid SIDQ req. within session (TotSessReq)

Total valid SIDQ requests within session

Requests for alternate contents (TotAltCntReq)

Total requests for alternate contents

In-Memory GET responses sent (TotGETInMemRsp)

Total in-memory GET responses sent from NS

In-Memory POST responses sent (TotPOSTInMemRsp)

Total in-memory POST responses sent from NS

In-Memory response bytes sent (TotInMemRspbytes)

Total in-memory response bytes sent from NS

appqoe CustomResp

The following operations can be performed on "appqoe CustomResp":

[import](#) | [rm](#) | [show](#) | [update](#)

import appqoe CustomResp

Downloads the input HTML Page to NetScaler Box with the given object name

Synopsys

```
import appqoe CustomResp [<src>] <name>
```

Arguments

src

name

Indicates name of the custom response HTML page to import/update.

Example

```
import appqoe CustomResp http://10.102.34.25/index.html appqoe_resp
```

rm appqoe CustomResp

Removes the imported HTML object.

Synopsys

```
rm appqoe CustomResp <name>
```

Arguments

name

Indicates name of the custom response HTML page to import/update.

Example

```
rm appqoe CustomResp appqoe_resp
```

show appqoe CustomResp

Displays lists all HTML page objects on the NetScaler appliance.

Synopsys

```
show appqoe CustomResp
```

Outputs

name

Indicates name of the custom response HTML page to import/update.

src

devno

count

stateflag

Example

```
show appqoe CustomResp
```

update appqoe CustomResp

Update the imported HTML object

Synopsys

```
update appqoe CustomResp <name>
```

Arguments

name

Indicates name of the custom response HTML page to import/update.

Example

```
update appqoe CustomResp appqoe_resp
```

appqoe action

The following operations can be performed on "appqoe action":

add | **rm** | **set** | **unset** | **show**

add appqoe action

Add a new AppQoE action for triggering

Synopsys

```
add appqoe action <name> [-priority <priority>] [-respondWith ( ACS | NS ) [<CustomFile>] [-altContentSvcName  
<string>] [-altContentPath <string>] [-maxConn <positive_integer>] [-delay <usecs>]] [-polqDepth <positive_integer>]  
[-priqDepth <positive_integer>] [-dosTrigExpression <expression>] [-dosAction ( SimpleResponse | HICResponse )]
```

Arguments

name

Name for the AppQoE action. Must begin with a letter, number, or the underscore symbol (_). Other characters allowed, after the first character, are the hyphen (-), period (.) hash (#), space (), at (@), equals (=), and colon (:) characters. This is a mandatory argument

priority

Priority for queuing the request. If server resources are not available for a request that matches the configured rule, this option specifies a priority for queuing the request until the server resources are available again. If priority is not configured then Lowest priority will be used to queue the request.

Possible values: HIGH, MEDIUM, LOW, LOWEST

respondWith

Responder action to be taken when the threshold is reached. Available settings function as follows:

ACS - Serve content from an alternative content service

Threshold : maxConn or delay

NS - Serve from the NetScaler appliance (built-in response)

Threshold : maxConn or delay

Possible values: ACS, NS

CustomFile

name of the HTML page object to use as the response

altContentSvcName

Name of the alternative content service to be used in the ACS

altContentPath

Path to the alternative content service to be used in the ACS

polqDepth

Policy queue depth threshold value. When the policy queue size (number of requests queued for the policy binding this action is attached to) increases to the specified polqDepth value, subsequent requests are dropped to the lowest priority level.

Minimum value: 0

Maximum value: 4294967294

priqDepth

Queue depth threshold value per priority level. If the queue size (number of requests in the queue of that particular priority) on the virtual server to which this policy is bound, increases to the specified qDepth value, subsequent requests are dropped to the lowest priority level.

Minimum value: 0

Maximum value: 4294967294

maxConn

Maximum number of concurrent connections that can be open for requests that matches with rule.

Minimum value: 1

Maximum value: 4294967294

delay

Delay threshold, in microseconds, for requests that match the policy's rule. If the delay statistics gathered for the matching request exceed the specified delay, configured action triggered for that request, if there is no action then requests are dropped to the lowest priority level

Minimum value: 1

Maximum value: 599999999

dosTrigExpression

Optional expression to add second level check to trigger DoS actions. Specifically used for Analytics based DoS response generation

dosAction

DoS Action to take when vserver will be considered under DoS attack and corresponding rule matches. Mandatory if AppQoE actions are to be used for DoS attack prevention.

Possible values: SimpleResponse, HICResponse

rm appqoe action

Removes the specified AppQoE action.

Synopsis

```
rm appqoe action <name>
```

Arguments

name

Name of the action to be removed.

set appqoe action

Set the argument of specified AppQoE action.

Synopsis

```
set appqoe action <name> [-priority <priority>] [-altContentSvcName <string>] [-altContentPath <string>] [-polqDepth <positive_integer>] [-priqDepth <positive_integer>] [-maxConn <positive_integer>] [-delay <usecs>] [-dosTrigExpression <expression>] [-dosAction ( SimpleResponse | HICResponse )]
```

Arguments

name

Name for the AppQoE action. Must begin with a letter, number, or the underscore symbol (_). Other characters allowed, after the first character, are the hyphen (-), period (.), hash (#), space (), at (@), equals (=), and colon (:) characters. This is a mandatory argument

priority

Priority for queuing the request. If server resources are not available for a request that matches the configured rule, this option specifies a priority for queuing the request until the server resources are available again. If priority is not configured then Lowest priority will be used to queue the request.

Possible values: HIGH, MEDIUM, LOW, LOWEST

altContentSvcName

Name of the alternative content service to be used in the ACS

altContentPath

Path to the alternative content service to be used in the ACS

polqDepth

Policy queue depth threshold value. When the policy queue size (number of requests queued for the policy binding this action is attached to) increases to the specified polqDepth value, subsequent requests are dropped to the lowest priority level.

Minimum value: 0

Maximum value: 4294967294

priqDepth

Queue depth threshold value per priority level. If the queue size (number of requests in the queue of that particular priority) on the virtual server to which this policy is bound, increases to the specified qDepth value, subsequent requests are dropped to the lowest priority level.

Minimum value: 0

Maximum value: 4294967294

maxConn

Maximum number of concurrent connections that can be open for requests that matches with rule.

Minimum value: 1

Maximum value: 4294967294

delay

Delay threshold, in microseconds, for requests that match the policy's rule. If the delay statistics gathered for the matching request exceed the specified delay, configured action triggered for that request, if there is no action then requests are dropped to the lowest priority level

Minimum value: 1

Maximum value: 599999999

dosTrigExpression

Optional expression to add second level check to trigger DoS actions. Specifically used for Analytics based DoS response generation

dosAction

DoS Action to take when vserver will be considered under DoS attack and corresponding rule matches. Mandatory if AppQoE actions are to be used for DoS attack prevention.

Possible values: SimpleResponse, HICResponse

unset appqoe action

Use this command to remove appqoe action settings. Refer to the set appqoe action command for meanings of the arguments.

Synopsys

```
unset appqoe action <name> [-priority] [-altContentSvcName] [-altContentPath] [-polqDepth] [-priqDepth] [-maxConn] [-delay] [-dosAction]
```

show appqoe action

Display configured AppQoE action(s).

Synopsys

```
show appqoe action [<name>]
```

Arguments

name

Name for the AppQoE action. Must begin with a letter, number, or the underscore symbol (_). Other characters allowed, after the first character, are the hyphen (-), period (.), hash (#), space (), at (@), equals (=), and colon (:) characters. This is a mandatory argument

Outputs

stateflag

hits

priority

Priority for queuing the request. If server resources are not available for a request that matches the configured rule, this option specifies a priority for queuing the request until the server resources are available again. If priority is not configured then Lowest priority will be used to queue the request.

respondWith

Responder action to be taken when the threshold is reached. Available settings function as follows:

ACS - Serve content from an alternative content service

Threshold : maxConn or delay

NS - Serve from the NetScaler appliance (built-in response)

Threshold : maxConn or delay

polqDepth

Policy queue depth threshold value. When the policy queue size (number of requests queued for the policy binding this action is attached to) increases to the specified polqDepth value, subsequent requests are dropped to the lowest priority level.

priqDepth

Queue depth threshold value per priority level. If the queue size (number of requests in the queue of that particular priority) on the virtual server to which this policy is bound, increases to the specified qDepth value, subsequent requests are dropped to the lowest priority level.

altContentSvcName

Name of the alternative content service to be used in the ACS

altContentPath

Path to the alternative content service to be used in the ACS

maxConn

Maximum number of concurrent connections that can be open for requests that matches with rule.

delay

Delay threshold, in microseconds, for requests that match the policy's rule. If the delay statistics gathered for the matching request exceed the specified delay, configured action triggered for that request, if there is no action then requests are dropped to the lowest priority level

CustomFile

name of the HTML page object to use as the response

dosTrigExpression

Optional expression to add second level check to trigger DoS actions. Specifically used for Analytics based DoS response generation

dosAction

DoS Action to take when vserver will be considered under DoS attack and corresponding rule matches. Mandatory if AppQoE actions are to be used for DoS attack prevention.

devno

count

appqoe parameter

The following operations can be performed on "appqoe parameter":

[set](#) | [unset](#) | [show](#)

set appqoe parameter

Sets the parameters for displaying appqoe information.

Synopsys

```
set appqoe parameter [-sessionLife <secs>] [-avgwaitingclient <positive_integer>] [-MaxAltRespBandWidth <positive_integer>] [-dosAttackThresh <positive_integer>]
```

Arguments

sessionLife

Time, in seconds, between the first time and the next time the AppQoE alternative content window is displayed. The alternative content window is displayed only once during a session for the same browser accessing a configured URL, so this parameter determines the length of a session.

Default value: 300

Minimum value: 1

Maximum value: 4294967294

avgwaitingclient

average number of client connections, that can sit in service waiting queue

Default value: 1000000

Minimum value: 0

Maximum value: 4294967294

MaxAltRespBandWidth

maximum bandwidth which will determine whether to send alternate content response

Default value: 100

Minimum value: 1

Maximum value: 4294967294

dosAttackThresh

average number of client connection that can queue up on vserver level without triggering DoS mitigation module

Default value: 2000

Minimum value: 0

Maximum value: 4294967294

Example

```
set appqoe parameter -sessionlife 200 -avgwaitingclient 10
```

unset appqoe parameter

Use this command to remove appqoe parameter settings. Refer to the set appqoe parameter command for meanings of the arguments.

Synopsys

unset appqoe parameter [-sessionLife] [-avgwaitingclient] [-MaxAltRespBandWidth] [-dosAttackThresh]

show appqoe parameter

Displays the values of the session life and filename parameters

Synopsys

show appqoe parameter

Outputs

sessionLife

appqoe session life (in seconds)

avgwaitingclient

average number of client connections, that can sit in service waiting queue

MaxAltRespBandWidth

maximum bandwidth which will determine whether to send alternate content response

dosAttackThresh

average number of client connection that can queue up on vserver level without triggering DoS mitigation module

Example

show appqos parameter

appqoe policy

The following operations can be performed on "appqoe policy":

add | **rm** | **set** | **show** | **stat**

add appqoe policy

Add a new AppQoE policy for binding rule with action

Synopsys

add appqoe policy <name> -rule <expression> -action <string>

Arguments

name

rule

Expression or name of a named expression, against which the request is evaluated. The policy is applied if the rule evaluates to true.

action

Configured AppQoE action to trigger

rm appqoe policy

Remove an AppQoE policy.

Synopsys

rm appqoe policy <name>

Arguments

name

Name of the AppQoE policy to be removed.

set appqoe policy

Synopsys

set appqoe policy <name> [-rule <expression>] [-action <string>]

Arguments

name

rule

Expression or name of a named expression, against which the request is evaluated. The policy is applied if the rule evaluates to true.

action

Configured AppQoE action to trigger

show appqoe policy

Display all the configured AppQoE policies.

Synopsys

show appqoe policy [<name>]

Arguments

name

Outputs

stateflag

rule

Expression or name of a named expression, against which the request is evaluated. The policy is applied if the rule evaluates to true.

action

Configured AppQoE action to trigger

hits

Number of hits.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

bindPriority

Specifies the binding of the policy. use only in display

boundTo

The name of the entity to which the policy is bound.

activePolicy

devno

count

stat appqoe policy

Displays collected brief statistics for all AppQoE policies, or detailed statistics for only the specified policy.

Synopsys

stat appqoe policy [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

name

policyName

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Server TTFB (SvrTTFB)

Server Time-To-First-Byte in milliseconds calculated for this AppQoE policy.

Server TTLB (SvrTTLB)

Server Time-To-Last-Byte in milliseconds calculated for this AppQoE policy.

Client TTLB (ClTTLB)

Client Time-To-Last-Byte in milliseconds calculated for this AppQoE policy.

Average Server TTFB (SvrTTFB)

Average Server Time-To-First-Byte in milliseconds calculated for this AppQoE policy.

Average Server TTLB (SvrTTLB)

Average Server Time-To-Last-Byte in milliseconds calculated for this AppQoE policy.

Average Client TTLB (ClTTLB)

Average Client Time-To-Last-Byte in milliseconds calculated for this AppQoE policy.

ThroughPut(Kbps) (ThroughPut)

Throughput in Kbps calculated on this AppQoE policy

Server TCP connections (TotSvr)

Total number of server connections that were established through this AppQoE Policy

Client TCP connections (TotClT)

Total number of client connections that were requested through this AppQoE Policy

Requests received (TotReq)

Total number of requests that were requested through this AppQoE policy

Requests bytes (TotReqBytes)

Total number of requests bytes that were requested through this AppQoE policy

Responses received (TotRsp)

Total number of responses received by this AppQoE policy

Response bytes (TotRspBytes)

Total number of response bytes received by this AppQoE policy

Alternate responses sent (TotJSsent)

Total number of in-memory responses sent instead of expected responses through this AppQoE policy

Alternate responses bytes sent (TotJSBytessent)

Total bytes of in-memory responses sent through this AppQoE policy

Policy hits (Hits)

Number of hits on the policy

Client HTTP transactions

Total number of client transactions processed by this AppQoE policy.

Svr HTTP transactions

Total number of server transactions processed by this AppQoE policy.

Example

```
stat appqos policy
```

appqoe stats

The following operations can be performed on "appqoe stats":

show appqoe stats

show appqoe stats is an alias for stat appqoe Displays global AppQoE statistics.

Synopsys

show appqoe stats - alias for 'stat appqoe'

Audit Commands

The entities on which you can perform NetScaler CLI operations:

- o audit
- o audit messageaction
- o audit messages
- o audit nslogAction
- o audit nslogParams
- o audit nslogPolicy
- o audit stats
- o audit syslogAction
- o audit syslogParams
- o audit syslogPolicy

audit

The following operations can be performed on "audit":

stat audit

Display the audit statistics

Synopsys

```
stat audit [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

Audit logs sent to syslog server(s) (LogSnd)

Syslog messages sent to the syslog server(s).

Audit log messages generated (LogGen)

Syslog messages about to be sent to the syslog server.

NAT allocation failed (Ernatpcb)

NAT allocation failed.

Nsb allocation failed (Ernsb)

Nsb allocation failed.

Memory allocation failed (Ermem)

Failures in allocation of Access Gateway context structure. When an Access Gateway session is established, the NetScaler creates an internal context structure , which identifies the user and the IP address from which the user has logged in.

Port allocation failed (Erport)

Number of times the NetScaler failed to allocate a port when sending a syslog message to the syslog server (s).

NAT lookup failed (Hshmiss)

NAT lookup failed.

Context not found (Ctxntfnd)

Failures in finding the context structure for an Access Gateway session during attempts to send session-specific audit messages.

During an Access Gateway session, audit messages related to the session are queued up in the auditlog buffer for transmission to the audit log server(s). If the session is killed before the messages are sent, the context structure allocated at session creation is removed. This structure is needed for sending the queued auditlog messages. If it is not found, this counter is incremented.

Nsb chain allocation failed (Ernsbchn)

Nsb Chain allocation failed.

Client connect failed (ErcIconn)

Failures in establishment of a connection between the NetScaler and the auditserver tool (the Netscaler's custom logging tool).

MP buffer flush command count (flcmdcnt)

Auditlog buffer flushes. In a multiprocessor NetScaler, both the main processor and the co-processor can generate auditlog messages and fill up the auditlog buffers. But only the primary processor can free up the buffers by sending auditlog messages to the auditlog server(s). The number of auditlog buffers is fixed. If the co-processor detects that all the auditlog buffers are full, it issues a flush command to the main processor.

audit messageaction

The following operations can be performed on "audit messageaction":

add | **rm** | **set** | **unset** | **show**

add audit messageaction

Adds an audit message action. The action specifies whether to log the message, and to which log.

Synopsys

```
add audit messageaction <name> <logLevel> <stringBuilderExpr> [-logtoNewslog ( YES | NO )] [-bypassSafetyCheck ( YES | NO )]
```

Arguments

name

Name of the audit message action. Must begin with a letter, number, or the underscore character (`_`), and must contain only letters, numbers, and the hyphen (`-`), period (`.`) pound (`#`), space (), at (`@`), equals (`=`), colon (`:`), and underscore characters. Cannot be changed after the message action is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, `?my message action?` or `'my message action?'`).

logLevel

Audit log level, which specifies the severity level of the log message being generated..

The following loglevels are valid:

- * EMERGENCY - Events that indicate an immediate crisis on the server.
- * ALERT - Events that might require action.
- * CRITICAL - Events that indicate an imminent server crisis.
- * ERROR - Events that indicate some type of error.
- * WARNING - Events that require action in the near future.
- * NOTICE - Events that the administrator should know about.
- * INFORMATIONAL - All but low-level events.
- * DEBUG - All events, in extreme detail.

Possible values: EMERGENCY, ALERT, CRITICAL, ERROR, WARNING, NOTICE, INFORMATIONAL, DEBUG

stringBuilderExpr

Default-syntax expression that defines the format and content of the log message.

logtoNewslog

Send the message to the new nslog.

Possible values: YES, NO

bypassSafetyCheck

Bypass the safety check and allow unsafe expressions.

Possible values: YES, NO

Default value: NO

rm audit messageaction

Removes the specified audit message action and associated configuration.

Synopsys

rm audit messageaction <name>

Arguments

name

Name of the audit message action to remove.

set audit messageaction

Modifies the specified parameters of an existing audit message action.

Synopsys

set audit messageaction <name> [-logLevel <logLevel>] [-stringBuilderExpr <string>] [-logtoNewnslog (YES | NO)]
[-bypassSafetyCheck (YES | NO)]

Arguments

name

Name of the audit message action to modify.

logLevel

Audit log level, which specifies the severity level of the log message being generated.

The following loglevels are valid:

- * EMERGENCY - Events that indicate an immediate crisis on the server.
- * ALERT - Events that might require action.
- * CRITICAL - Events that indicate an imminent server crisis.
- * ERROR - Events that indicate some type of error.
- * WARNING - Events that require action in the near future.
- * NOTICE - Events that the administrator should know about.
- * INFORMATIONAL - All but low-level events.
- * DEBUG - All events, in extreme detail.

Possible values: EMERGENCY, ALERT, CRITICAL, ERROR, WARNING, NOTICE, INFORMATIONAL, DEBUG

stringBuilderExpr

Default-syntax expression that defines the format and content of the log message.

logtoNewnslog

Send the message to the new nslog.

Possible values: YES, NO

bypassSafetyCheck

Bypass the safety check and allow unsafe expressions.

Possible values: YES, NO

Default value: NO

unset audit messageaction

Use this command to remove audit messageaction settings. Refer to the set audit messageaction command for meanings of the arguments.

Synopsys

```
unset audit messageaction <name> [-logtoNewslog] [-bypassSafetyCheck]
```

show audit messageaction

Displays the current configuration of the specified audit message action. If no audit message action is specified, displays a list of all audit message actions currently configured on the NetScaler appliance.

Synopsys

```
show audit messageaction [<name>]
```

Arguments

name

Name of the audit message action.

Outputs

logLevel

stringBuilderExpr

Default-syntax expression that defines the format and content of the log message.

logtoNewslog

Send the message to the new nslog.

bypassSafetyCheck

Bypass the safety check and allow unsafe expressions.

stateflag

hits

The number of times the action has been taken.

undefHits

The number of times the action resulted in UNDEF.

referenceCount

The number of references to the action.

devno

count

audit messages

The following operations can be performed on "audit messages":

show audit messages

Displays the most recent audit log messages.

Synopsys

show audit messages [-logLevel <logLevel> ...] [-numOfMesgs <positive_integer>]

Arguments

logLevel

Audit log level filter, which specifies the types of events to display.

The following loglevels are valid:

- * ALL - All events.
- * EMERGENCY - Events that indicate an immediate crisis on the server.
- * ALERT - Events that might require action.
- * CRITICAL - Events that indicate an imminent server crisis.
- * ERROR - Events that indicate some type of error.
- * WARNING - Events that require action in the near future.
- * NOTICE - Events that the administrator should know about.
- * INFORMATIONAL - All but low-level events.
- * DEBUG - All events, in extreme detail.

numOfMesgs

Number of log messages to be displayed.

Default value: 20

Minimum value: 1

Maximum value: 256

Outputs

value

The Audit message

devno

count

stateflag

audit nslogAction

The following operations can be performed on "audit nslogAction":

add | **rm** | **set** | **unset** | **show**

add audit nslogAction

Adds an nslog action. The action contains a reference to an nslog server and specifies which information to log and how to log that information.

Synopsys

```
add audit nslogAction <name> <serverIP> [-serverPort <port>] -logLevel <logLevel> ... [-dateFormat <dateFormat>]
[-logFacility <logFacility>] [-tcp ( NONE | ALL )] [-acl ( ENABLED | DISABLED )] [-timeZone ( GMT_TIME |
LOCAL_TIME )] [-userDefinedAuditlog ( YES | NO )] [-appflowExport ( ENABLED | DISABLED )]
```

Arguments

name

Name of the nslog action. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space (), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after the nslog action is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my nslog action? or ?my nslog action).

serverIP

IP address of the nslog server.

serverPort

Port on which the nslog server accepts connections.

Minimum value: 1

logLevel

Audit log level, which specifies the types of events to log.

Available settings function as follows:

- * ALL - All events.
- * EMERGENCY - Events that indicate an immediate crisis on the server.
- * ALERT - Events that might require action.
- * CRITICAL - Events that indicate an imminent server crisis.
- * ERROR - Events that indicate some type of error.
- * WARNING - Events that require action in the near future.
- * NOTICE - Events that the administrator should know about.
- * INFORMATIONAL - All but low-level events.
- * DEBUG - All events, in extreme detail.
- * NONE - No events.

dateFormat

Format of dates in the logs.

Supported formats are:

- * MMDDYYYY - U.S. style month/date/year format.

- * DDMMYYYY - European style date/month/year format.

- * YYYYMMDD - ISO style year/month/date format.

Possible values: MMDDYYYY, DDMMYYYY, YYYYMMDD

logFacility

Facility value, as defined in RFC 3164, assigned to the log message.

Log facility values are numbers 0 to 7 (LOCAL0 through LOCAL7). Each number indicates where a specific message originated from, such as the NetScaler appliance itself, the VPN, or external.

Possible values: LOCAL0, LOCAL1, LOCAL2, LOCAL3, LOCAL4, LOCAL5, LOCAL6, LOCAL7

tcp

Log TCP messages.

Possible values: NONE, ALL

acl

Log access control list (ACL) messages.

Possible values: ENABLED, DISABLED

timeZone

Time zone used for date and timestamps in the logs.

Available settings function as follows:

- * GMT_TIME. Coordinated Universal Time.

- * LOCAL_TIME. The server's timezone setting.

Possible values: GMT_TIME, LOCAL_TIME

userDefinedAuditlog

Log user-configurable log messages to nslog.

Setting this parameter to NO causes auditing to ignore all user-configured message actions. Setting this parameter to YES causes auditing to log user-configured message actions that meet the other logging criteria.

Possible values: YES, NO

appflowExport

Export log messages to AppFlow collectors.

Appflow collectors are entities to which log messages can be sent so that some action can be performed on them.

Possible values: ENABLED, DISABLED

rm audit nslogAction

Removes the specified nslog action and associated configuration. Note: An nslog action cannot be removed if it is bound to an nslog policy.

Synopsys

rm audit nslogAction <name>

Arguments

name

Name of the nslog action to remove.

set audit nslogAction

Modifies the specified settings of an existing nslog action.

Synopsis

```
set audit nslogAction <name> [-serverIP <ip_addr|ipv6_addr|*>] [-serverPort <port>] [-logLevel <logLevel> ...] [-dateFormat <dateFormat>] [-logFacility <logFacility>] [-tcp ( NONE | ALL )] [-acl ( ENABLED | DISABLED )] [-timeZone ( GMT_TIME | LOCAL_TIME )] [-userDefinedAuditlog ( YES | NO )] [-appflowExport ( ENABLED | DISABLED )]
```

Arguments

name

Name of the nslog action to be modified.

serverIP

IP address of the nslog server.

serverPort

Port on which the nslog server accepts connections.

Minimum value: 1

logLevel

Audit log level, which specifies the types of events to log.

Available settings function as follows:

- * ALL - All events.
- * EMERGENCY - Events that indicate an immediate crisis on the server.
- * ALERT - Events that might require action.
- * CRITICAL - Events that indicate an imminent server crisis.
- * ERROR - Events that indicate some type of error.
- * WARNING - Events that require action in the near future.
- * NOTICE - Events that the administrator should know about.
- * INFORMATIONAL - All but low-level events.
- * DEBUG - All events, in extreme detail.
- * NONE - No events.

dateFormat

Format of dates in the logs.

Supported formats are:

- * MMDDYYYY - U.S. style month/date/year format.

* DDMMYYYY - European style date/month/year format.

* YYYYMMDD - ISO style year/month/date format.

Possible values: MMDDYYYY, DDMMYYYY, YYYYMMDD

logFacility

Facility value, as defined in RFC 3164, assigned to the log message.

Log facility values are numbers 0 to 7 (LOCAL0 through LOCAL7). Each number indicates where a specific message originated from, such as the NetScaler appliance itself, the VPN, or external.

Possible values: LOCAL0, LOCAL1, LOCAL2, LOCAL3, LOCAL4, LOCAL5, LOCAL6, LOCAL7

tcp

Log TCP messages.

Possible values: NONE, ALL

acl

Log access control list (ACL) messages.

Possible values: ENABLED, DISABLED

timeZone

Time zone used for date and timestamps in the logs.

Available settings function as follows:

* GMT_TIME. Coordinated Universal Time.

* LOCAL_TIME. The server's timezone setting.

Possible values: GMT_TIME, LOCAL_TIME

userDefinedAuditlog

Log user-configurable log messages to nslog.

Setting this parameter to NO causes auditing to ignore all user-configured message actions. Setting this parameter to YES causes auditing to log user-configured message actions that meet the other logging criteria.

Possible values: YES, NO

appflowExport

Export log messages to AppFlow collectors.

Appflow collectors are entities to which log messages can be sent so that some action can be performed on them.

Possible values: ENABLED, DISABLED

unset audit nslogAction

Removes the settings of an existing nslog action. Attributes for which a default value is available revert to their default values. See the set audit nslogAction command for descriptions of the parameters..Refer to the set audit nslogAction command for meanings of the arguments.

Synopsis

```
unset audit nslogAction <name> [-serverPort] [-logLevel] [-dateFormat] [-logFacility] [-tcp] [-acl] [-timeZone] [-userDefinedAuditlog] [-appflowExport]
```

show audit nslogAction

Displays the current configuration of the specified nslog action. If no nslog action is specified, displays a list of all nslog actions currently configured on the NetScaler appliance.

Synopsys

show audit nslogAction [<name>]

Arguments

name

Name of the nslog action.

Outputs

serverIP

IP address of the nslog server.

serverPort

Port on which the nslog server accepts connections.

logLevel

Audit log level, which specifies the types of events to log.

Available settings function as follows:

- * ALL - All events.
- * EMERGENCY - Events that indicate an immediate crisis on the server.
- * ALERT - Events that might require action.
- * CRITICAL - Events that indicate an imminent server crisis.
- * ERROR - Events that indicate some type of error.
- * WARNING - Events that require action in the near future.
- * NOTICE - Events that the administrator should know about.
- * INFORMATIONAL - All but low-level events.
- * DEBUG - All events, in extreme detail.
- * NONE - No events.

dateFormat

Format of dates in the logs.

Supported formats are:

- * MMDDYYYY - U.S. style month/date/year format.
- * DDMMYYYY - European style date/month/year format.
- * YYYYMMDD - ISO style year/month/date format.

logFacility

Facility value, as defined in RFC 3164, assigned to the log message.

Log facility values are numbers 0 to 7 (LOCAL0 through LOCAL7). Each number indicates where a specific message originated from, such as the NetScaler appliance itself, the VPN, or external.

tcp

Log TCP messages.

acl

Log access control list (ACL) messages.

timeZone

Time zone used for date and timestamps in the logs.

Available settings function as follows:

- * GMT_TIME. Coordinated Universal Time.
- * LOCAL_TIME. The server's timezone setting.

stateflag**userDefinedAuditlog**

Log user-configurable log messages to nslog.

Setting this parameter to NO causes auditing to ignore all user-configured message actions. Setting this parameter to YES causes auditing to log user-configured message actions that meet the other logging criteria.

appflowExport

Export log messages to AppFlow collectors.

Appflow collectors are entities to which log messages can be sent so that some action can be performed on them.

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

devno**count**

audit nslogParams

The following operations can be performed on "audit nslogParams":

[set](#) | [unset](#) | [show](#)

set audit nslogParams

Modifies the specified nslog parameters. Changes the IP address, the port, or the logging parameters for logs sent to nslog.

Synopsys

```
set audit nslogParams [-serverIP <ip_addr|ipv6_addr|*>] [-serverPort <port>] [-dateFormat <dateFormat>] [-logLevel <logLevel> ...] [-logFacility <logFacility>] [-tcp ( NONE | ALL )] [-acl ( ENABLED | DISABLED )] [-timeZone ( GMT_TIME | LOCAL_TIME )] [-userDefinedAuditlog ( YES | NO )] [-appflowExport ( ENABLED | DISABLED )]
```

Arguments

serverIP

IP address of the nslog server.

serverPort

Port on which the nslog server accepts connections.

Minimum value: 1

dateFormat

Format of dates in the logs.

Supported formats are:

* MMDDYYYY - U.S. style month/date/year format.

* DDMMYYYY - European style date/month/year format.

* YYYYMMDD - ISO style year/month/date format.

Possible values: MMDDYYYY, DDMMYYYY, YYYYMMDD

logLevel

Types of information to be logged.

Available settings function as follows:

* ALL - All events.

* EMERGENCY - Events that indicate an immediate crisis on the server.

* ALERT - Events that might require action.

* CRITICAL - Events that indicate an imminent server crisis.

* ERROR - Events that indicate some type of error.

* WARNING - Events that require action in the near future.

* NOTICE - Events that the administrator should know about.

* INFORMATIONAL - All but low-level events.

* DEBUG - All events, in extreme detail.

* NONE - No events.

logFacility

Facility value, as defined in RFC 3164, assigned to the log message.

Log facility values are numbers 0 to 7 (LOCAL0 through LOCAL7). Each number indicates where a specific message originated from, such as the NetScaler appliance itself, the VPN, or external.

Possible values: LOCAL0, LOCAL1, LOCAL2, LOCAL3, LOCAL4, LOCAL5, LOCAL6, LOCAL7

tcp

Configure auditing to log TCP messages.

Possible values: NONE, ALL

acl

Configure auditing to log access control list (ACL) messages.

Possible values: ENABLED, DISABLED

timeZone

Time zone used for date and timestamps in the logs.

Supported settings are:

* GMT_TIME - Coordinated Universal Time.

* LOCAL_TIME - Use the server's timezone setting.

Possible values: GMT_TIME, LOCAL_TIME

userDefinedAuditlog

Log user-configurable log messages to nslog.

Setting this parameter to NO causes auditing to ignore all user-configured message actions. Setting this parameter to YES causes auditing to log user-configured message actions that meet the other logging criteria.

Possible values: YES, NO

appflowExport

Export log messages to AppFlow collectors.

Appflow collectors are entities to which log messages can be sent so that some action can be performed on them.

Possible values: ENABLED, DISABLED

unset audit nslogParams

Removes the existing nslog parameter settings. Attributes for which a default value is available revert to their default values. See the set audit nslogParams command for a description of the parameters..Refer to the set audit nslogParams command for meanings of the arguments.

Synopsys

```
unset audit nslogParams [-serverIP] [-serverPort] [-logLevel] [-dateFormat] [-logFacility] [-tcp] [-acl] [-timeZone] [-userDefinedAuditlog] [-appflowExport]
```

show audit nslogParams

Displays the current nslog parameter settings.

Synopsys

Outputs

name

Name of the nslog param.

serverIP

IP address of the nslog server.

serverPort

Port on which the nslog server accepts connections.

dateFormat

Format of dates in the logs.

Supported formats are:

- * MMDDYYYY - U.S. style month/date/year format.
- * DDMMYYYY - European style date/month/year format.
- * YYYYMMDD - ISO style year/month/date format.

logLevel

The audit log level.

logFacility

Facility value, as defined in RFC 3164, assigned to the log message.

Log facility values are numbers 0 to 7 (LOCAL0 through LOCAL7). Each number indicates where a specific message originated from, such as the NetScaler appliance itself, the VPN, or external.

tcp

Configure auditing to log TCP messages.

acl

Configure auditing to log access control list (ACL) messages.

timeZone

Time zone used for date and timestamps in the logs.

Supported settings are:

- * GMT_TIME - Coordinated Universal Time.
- * LOCAL_TIME - Use the server's timezone setting.

userDefinedAuditlog

Log user-configurable log messages to nslog.

Setting this parameter to NO causes auditing to ignore all user-configured message actions. Setting this parameter to YES causes auditing to log user-configured message actions that meet the other logging criteria.

appflowExport

Export log messages to AppFlow collectors.

Appflow collectors are entities to which log messages can be sent so that some action can be performed on them.

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

audit nslogPolicy

The following operations can be performed on "audit nslogPolicy":

[add](#) | [rm](#) | [set](#) | [show](#)

add audit nslogPolicy

Adds a policy that defines which messages to log to the specified nslog server.

Synopsis

```
add audit nslogPolicy <name> <rule> <action>
```

Arguments

name

Name for the policy.

Must begin with a letter, number, or the underscore character (`_`), and must consist only of letters, numbers, and the hyphen (`-`), period (`.`) pound (`#`), space (), at sign (`@`), equals (`=`), colon (`:`), and underscore characters. Cannot be changed after the nslog policy is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, `?my nslog policy?` or `'my nslog policy'`).

rule

Name of the NetScaler named rule, or a default syntax expression, that defines the messages to be logged to the nslog server.

action

Nslog server action that is performed when this policy matches.

NOTE: An nslog server action must be associated with an nslog audit policy.

rm audit nslogPolicy

Removes the specified nslog policy and associated configuration.

Synopsis

```
rm audit nslogPolicy <name>
```

Arguments

name

Name of the nslog policy to remove.

set audit nslogPolicy

Modifies the specified parameters of an existing nslog policy.

Synopsis

```
set audit nslogPolicy <name> [-rule <expression>] [-action <string>]
```

Arguments

name

Name of the nslog policy to modify.

rule

Name of the NetScaler named rule, or a default syntax expression, that defines the messages to be logged to the nslog server.

action

Nslog server action that is performed when this policy matches.

NOTE: An nslog server action must be associated with an nslog audit policy.

show audit nslogPolicy

Displays the current configuration of the specified nslog policy. If no nslog policy is specified, displays a list of all nslog policies currently configured on the NetScaler appliance.

Synopsys

show audit nslogPolicy [<name>]

Arguments

name

Name of the policy.

Outputs

rule

Name of the NetScaler named rule, or a default syntax expression, that defines the messages to be logged to the nslog server.

action

Nslog server action that is performed when this policy matches.

NOTE: An nslog server action must be associated with an nslog audit policy.

boundTo

The entity name to which policy is bound

activePolicy**priority****bindPolicyType****policyType****builtin**

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

devno**count****stateflag**

audit stats

The following operations can be performed on "audit stats":

show audit stats

show audit stats is an alias for stat audit

Synopsys

show audit stats - alias for 'stat audit'

audit syslogAction

The following operations can be performed on "audit syslogAction":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add audit syslogAction

Adds a syslog action. The action contains a reference to a syslog server, and specifies which information to log and how to log that information.

Synopsys

```
add audit syslogAction <name> <serverIP> [-serverPort <port>] -logLevel <logLevel> ... [-dateFormat <dateFormat>]
[-logFacility <logFacility>] [-tcp ( NONE | ALL )] [-acl ( ENABLED | DISABLED )] [-timeZone ( GMT_TIME |
LOCAL_TIME )] [-userDefinedAuditlog ( YES | NO )] [-appflowExport ( ENABLED | DISABLED )]
```

Arguments

name

Name of the syslog action. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space (), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after the syslog action is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my syslog action" or 'my syslog action').

serverIP

IP address of the syslog server.

serverPort

Port on which the syslog server accepts connections.

Minimum value: 1

logLevel

Audit log level, which specifies the types of events to log.

Available values function as follows:

- * ALL - All events.
- * EMERGENCY - Events that indicate an immediate crisis on the server.
- * ALERT - Events that might require action.
- * CRITICAL - Events that indicate an imminent server crisis.
- * ERROR - Events that indicate some type of error.
- * WARNING - Events that require action in the near future.
- * NOTICE - Events that the administrator should know about.
- * INFORMATIONAL - All but low-level events.
- * DEBUG - All events, in extreme detail.
- * NONE - No events.

dateFormat

Format of dates in the logs.

Supported formats are:

* MMDDYYYY. -U.S. style month/date/year format.

* DDMMYYYY - European style date/month/year format.

* YYYYMMDD - ISO style year/month/date format.

Possible values: MMDDYYYY, DDMMYYYY, YYYYMMDD

logFacility

Facility value, as defined in RFC 3164, assigned to the log message.

Log facility values are numbers 0 to 7 (LOCAL0 through LOCAL7). Each number indicates where a specific message originated from, such as the NetScaler appliance itself, the VPN, or external.

Possible values: LOCAL0, LOCAL1, LOCAL2, LOCAL3, LOCAL4, LOCAL5, LOCAL6, LOCAL7

tcp

Log TCP messages.

Possible values: NONE, ALL

acl

Log access control list (ACL) messages.

Possible values: ENABLED, DISABLED

timeZone

Time zone used for date and timestamps in the logs.

Supported settings are:

* GMT_TIME. Coordinated Universal time.

* LOCAL_TIME. Use the server's timezone setting.

Possible values: GMT_TIME, LOCAL_TIME

userDefinedAuditlog

Log user-configurable log messages to syslog.

Setting this parameter to NO causes auditing to ignore all user-configured message actions. Setting this parameter to YES causes auditing to log user-configured message actions that meet the other logging criteria.

Possible values: YES, NO

appflowExport

Export log messages to AppFlow collectors.

Appflow collectors are entities to which log messages can be sent so that some action can be performed on them.

Possible values: ENABLED, DISABLED

rm audit syslogAction

Removes the specified syslog action and associated configuration. Note: A syslog action cannot be removed if it is bound to a syslog policy.

Synopsys

rm audit syslogAction <name>

Arguments

name

Name of the syslog action to remove.

set audit syslogAction

Modifies the specified parameters of an existing syslog action.

Synopsis

```
set audit syslogAction <name> [-serverIP <ip_addr|ipv6_addr*>] [-serverPort <port>] [-logLevel <logLevel> ...] [-dateFormat <dateFormat>] [-logFacility <logFacility>] [-tcp ( NONE | ALL )] [-acl ( ENABLED | DISABLED )] [-timeZone ( GMT_TIME | LOCAL_TIME )] [-userDefinedAuditlog ( YES | NO )] [-appflowExport ( ENABLED | DISABLED )]
```

Arguments

name

Name of the syslog action to be modified.

serverIP

IP address of the syslog server.

serverPort

Port on which the syslog server accepts connections.

Minimum value: 1

logLevel

Audit log level, which specifies the types of events to log.

Available values function as follows:

- * ALL - All events.
- * EMERGENCY - Events that indicate an immediate crisis on the server.
- * ALERT - Events that might require action.
- * CRITICAL - Events that indicate an imminent server crisis.
- * ERROR - Events that indicate some type of error.
- * WARNING - Events that require action in the near future.
- * NOTICE - Events that the administrator should know about.
- * INFORMATIONAL - All but low-level events.
- * DEBUG - All events, in extreme detail.
- * NONE - No events.

dateFormat

Format of dates in the logs.

Supported formats are:

- * MMDDYYYY. -U.S. style month/date/year format.

* DDMMYYYY - European style date/month/year format.

* YYYYMMDD - ISO style year/month/date format.

Possible values: MMDDYYYY, DDMMYYYY, YYYYMMDD

logFacility

Facility value, as defined in RFC 3164, assigned to the log message.

Log facility values are numbers 0 to 7 (LOCAL0 through LOCAL7). Each number indicates where a specific message originated from, such as the NetScaler appliance itself, the VPN, or external.

Possible values: LOCAL0, LOCAL1, LOCAL2, LOCAL3, LOCAL4, LOCAL5, LOCAL6, LOCAL7

tcp

Log TCP messages.

Possible values: NONE, ALL

acl

Log access control list (ACL) messages.

Possible values: ENABLED, DISABLED

timeZone

Time zone used for date and timestamps in the logs.

Supported settings are:

* GMT_TIME. Coordinated Universal time.

* LOCAL_TIME. Use the server's timezone setting.

Possible values: GMT_TIME, LOCAL_TIME

userDefinedAuditlog

Log user-configurable log messages to syslog.

Setting this parameter to NO causes auditing to ignore all user-configured message actions. Setting this parameter to YES causes auditing to log user-configured message actions that meet the other logging criteria.

Possible values: YES, NO

appflowExport

Export log messages to AppFlow collectors.

Appflow collectors are entities to which log messages can be sent so that some action can be performed on them.

Possible values: ENABLED, DISABLED

unset audit syslogAction

Removes the settings of an existing syslog action. Attributes for which a default value is available revert to their default values. See the set audit syslogAction command for a description of the parameters..Refer to the set audit syslogAction command for meanings of the arguments.

Synopsis

```
unset audit syslogAction <name> [-serverPort] [-logLevel] [-dateFormat] [-logFacility] [-tcp] [-acl] [-timeZone] [-userDefinedAuditlog] [-appflowExport] [-serverIP]
```


show audit syslogAction

Displays the current configuration of the specified syslog action. If no syslog action is specified, displays a list of all syslog actions currently configured on the NetScaler appliance.

Synopsys

show audit syslogAction [<name>]

Arguments

name

Name of the syslog action.

Outputs

serverIP

IP address of the syslog server.

serverPort

Port on which the syslog server accepts connections.

logLevel

Audit log level, which specifies the types of events to log.

Available values function as follows:

- * ALL - All events.
- * EMERGENCY - Events that indicate an immediate crisis on the server.
- * ALERT - Events that might require action.
- * CRITICAL - Events that indicate an imminent server crisis.
- * ERROR - Events that indicate some type of error.
- * WARNING - Events that require action in the near future.
- * NOTICE - Events that the administrator should know about.
- * INFORMATIONAL - All but low-level events.
- * DEBUG - All events, in extreme detail.
- * NONE - No events.

dateFormat

Format of dates in the logs.

Supported formats are:

- * MMDDYYYY. -U.S. style month/date/year format.
- * DDMMYYYY - European style date/month/year format.
- * YYYYMMDD - ISO style year/month/date format.

logFacility

Facility value, as defined in RFC 3164, assigned to the log message.

Log facility values are numbers 0 to 7 (LOCAL0 through LOCAL7). Each number indicates where a specific message originated from, such as the NetScaler appliance itself, the VPN, or external.

tcp

Log TCP messages.

acl

Log access control list (ACL) messages.

timeZone

Time zone used for date and timestamps in the logs.

Supported settings are:

- * GMT_TIME. Coordinated Universal time.
- * LOCAL_TIME. Use the server's timezone setting.

stateflag**userDefinedAuditlog**

Log user-configurable log messages to syslog.

Setting this parameter to NO causes auditing to ignore all user-configured message actions. Setting this parameter to YES causes auditing to log user-configured message actions that meet the other logging criteria.

appflowExport

Disable export of log messages to AppFlow collectors.

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

devno**count**

audit syslogParams

The following operations can be performed on "audit syslogParams":

[set](#) | [unset](#) | [show](#)

set audit syslogParams

Modifies the syslog parameters. Changes the IP, the port, or the logging parameters for logs sent to syslog.

Synopsis

```
set audit syslogParams [-serverIP <ip_addr|ipv6_addr|*>] [-serverPort <port>] [-dateFormat <dateFormat>] [-logLevel <logLevel> ...] [-logFacility <logFacility>] [-tcp ( NONE | ALL )] [-acl ( ENABLED | DISABLED )] [-timeZone ( GMT_TIME | LOCAL_TIME )] [-userDefinedAuditlog ( YES | NO )] [-appflowExport ( ENABLED | DISABLED )]
```

Arguments

serverIP

IP address of the syslog server.

serverPort

Port on which the syslog server accepts connections.

Minimum value: 1

dateFormat

Format of dates in the logs.

Supported formats are:

- * MMDDYYYY - U.S. style month/date/year format.
- * DDMMYYYY. European style -date/month/year format.
- * YYYYMMDD - ISO style year/month/date format.

Possible values: MMDDYYYY, DDMMYYYY, YYYYMMDD

logLevel

Types of information to be logged.

Available settings function as follows:

- * ALL - All events.
- * EMERGENCY - Events that indicate an immediate crisis on the server.
- * ALERT - Events that might require action.
- * CRITICAL - Events that indicate an imminent server crisis.
- * ERROR - Events that indicate some type of error.
- * WARNING - Events that require action in the near future.
- * NOTICE - Events that the administrator should know about.
- * INFORMATIONAL - All but low-level events.
- * DEBUG - All events, in extreme detail.
- * NONE - No events.

logFacility

Facility value, as defined in RFC 3164, assigned to the log message.

Log facility values are numbers 0 to 7 (LOCAL0 through LOCAL7). Each number indicates where a specific message originated from, such as the NetScaler appliance itself, the VPN, or external.

Possible values: LOCAL0, LOCAL1, LOCAL2, LOCAL3, LOCAL4, LOCAL5, LOCAL6, LOCAL7

tcp

Log TCP messages.

Possible values: NONE, ALL

acl

Log access control list (ACL) messages.

Possible values: ENABLED, DISABLED

timeZone

Time zone used for date and timestamps in the logs.

Available settings function as follows:

* GMT_TIME - Coordinated Universal Time.

* LOCAL_TIME Use the server's timezone setting.

Possible values: GMT_TIME, LOCAL_TIME

userDefinedAuditlog

Log user-configurable log messages to syslog.

Setting this parameter to NO causes audit to ignore all user-configured message actions. Setting this parameter to YES causes audit to log user-configured message actions that meet the other logging criteria.

Possible values: YES, NO

appflowExport

Export log messages to AppFlow collectors.

Appflow collectors are entities to which log messages can be sent so that some action can be performed on them.

Possible values: ENABLED, DISABLED

unset audit syslogParams

Removes the existing syslog parameter settings. Attributes for which a default value is available revert to their default values. See the set audit syslogParams command for descriptions of the parameters..Refer to the set audit syslogParams command for meanings of the arguments.

Synopsis

```
unset audit syslogParams [-serverIP] [-serverPort] [-logLevel] [-dateFormat] [-logFacility] [-tcp] [-acl] [-timeZone] [-userDefinedAuditlog] [-appflowExport]
```

show audit syslogParams

Displays the current syslog parameter settings.

Synopsis

Outputs

name

Name.

serverIP

IP address of the syslog server.

serverPort

Port on which the syslog server accepts connections.

dateFormat

Format of dates in the logs.

Supported formats are:

- * MMDDYYYY - U.S. style month/date/year format.
- * DDMMYYYY. European style -date/month/year format.
- * YYYYMMDD - ISO style year/month/date format.

logLevel

Types of information to be logged.

Available settings function as follows:

- * ALL - All events.
- * EMERGENCY - Events that indicate an immediate crisis on the server.
- * ALERT - Events that might require action.
- * CRITICAL - Events that indicate an imminent server crisis.
- * ERROR - Events that indicate some type of error.
- * WARNING - Events that require action in the near future.
- * NOTICE - Events that the administrator should know about.
- * INFORMATIONAL - All but low-level events.
- * DEBUG - All events, in extreme detail.
- * NONE - No events.

logFacility

Facility value, as defined in RFC 3164, assigned to the log message.

Log facility values are numbers 0 to 7 (LOCAL0 through LOCAL7). Each number indicates where a specific message originated from, such as the NetScaler appliance itself, the VPN, or external.

tcp

Log TCP messages.

acl

Log access control list (ACL) messages.

timeZone

Time zone used for date and timestamps in the logs.

Available settings function as follows:

- * GMT_TIME - Coordinated Universal Time.

- * LOCAL_TIME Use the server's timezone setting.

userDefinedAuditlog

Log user-configurable log messages to syslog.

Setting this parameter to NO causes audit to ignore all user-configured message actions. Setting this parameter to YES causes audit to log user-configured message actions that meet the other logging criteria.

appflowExport

Export log messages to AppFlow collectors.

Appflow collectors are entities to which log messages can be sent so that some action can be performed on them.

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

audit syslogPolicy

The following operations can be performed on "audit syslogPolicy":

[add](#) | [rm](#) | [set](#) | [show](#)

add audit syslogPolicy

Adds a policy that defines which messages to log to the specified syslog server.

Synopsys

add audit syslogPolicy <name> <rule> <action>

Arguments

name

Name for the policy.

Must begin with a letter, number, or the underscore character (_), and must consist only of letters, numbers, and the hyphen (-), period (.) pound (#), space (), at sign (@), equals (=), colon (:), and underscore characters. Cannot be changed after the syslog policy is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my syslog policy? or ?my syslog policy).

rule

Name of the NetScaler named rule, or a default syntax expression, that defines the messages to be logged to the syslog server.

action

Syslog server action to perform when this policy matches traffic.

NOTE: A syslog server action must be associated with a syslog audit policy.

rm audit syslogPolicy

Removes the specified syslog policy and associated configuration.

Synopsys

rm audit syslogPolicy <name>

Arguments

name

Name of the syslog policy to remove.

set audit syslogPolicy

Configures an existing syslog policy.

Synopsys

set audit syslogPolicy <name> [-rule <expression>] [-action <string>]

Arguments

name

Name of the syslog policy to be configured.

rule

Name of the NetScaler named rule, or a default syntax expression, that defines the messages to be logged to the syslog server.

action

Syslog server action to perform when this policy matches traffic.

NOTE: A syslog server action must be associated with a syslog audit policy.

show audit syslogPolicy

Displays the current configuration of the specified syslog policy. If no syslog policy is specified, displays a list of all syslog policies currently configured on the NetScaler appliance.

Synopsys

show audit syslogPolicy [<name>]

Arguments

name

Name of the policy.

Outputs

rule

Name of the NetScaler named rule, or a default syntax expression, that defines the messages to be logged to the syslog server.

action

Syslog server action to perform when this policy matches traffic.

NOTE: A syslog server action must be associated with a syslog audit policy.

boundTo

The entity name to which policy is bound

activePolicy**priority****bindPolicyType****policyType****builtin**

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

devno**count****stateflag**

Authentication Commands

The entities on which you can perform NetScaler CLI operations:

- o authentication Policy
- o authentication authnProfile
- o authentication certAction
- o authentication certPolicy
- o authentication ldapAction
- o authentication ldapPolicy
- o authentication localPolicy
- o authentication negotiateAction
- o authentication negotiatePolicy
- o authentication policylabel
- o authentication radiusAction
- o authentication radiusPolicy
- o authentication samlAction
- o authentication samlIdPPolicy
- o authentication samlIdPProfile
- o authentication samlPolicy
- o authentication tacacsAction
- o authentication tacacsPolicy
- o authentication vserver
- o authentication webAuthAction
- o authentication webAuthPolicy

authentication Policy

The following operations can be performed on "authentication Policy":

add | **rm** | **set** | **unset** | **show** | **rename** | **stat**

add authentication Policy

Adds an advanced authentication policy. The policy defines the criteria under which the NetScaler appliance attempts to authenticate the user.

Synopsys

```
add authentication Policy <name> -rule <expression> -action <string> [-undefAction <string>] [-comment <string>] [-logAction <string>]
```

Arguments

name

Name for the advance AUTHENTICATION policy.

Must begin with a letter, number, or the underscore character (`_`), and must contain only letters, numbers, and the hyphen (`-`), period (`.`) pound (`#`), space (), at (`@`), equals (`=`), colon (`:`), and underscore characters. Cannot be changed after AUTHENTICATION policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my authentication policy" or 'my authentication policy').

rule

Name of the NetScaler named rule, or a default syntax expression, that the policy uses to determine whether to attempt to authenticate the user with the AUTHENTICATION server.

action

Name of the authentication action to be performed if the policy matches.

undefAction

Action to perform if the result of policy evaluation is undefined (UNDEF). An UNDEF event indicates an internal error condition. Only the above built-in actions can be used.

comment

Any comments to preserve information about this policy.

logAction

Name of message log action to use when a request matches this policy.

rm authentication Policy

Removes the advance authentication policy.

Synopsys

```
rm authentication Policy <name>
```

Arguments

name

Name of the advance authentication policy to remove.

set authentication Policy

Modifies the specified parameters of a authentication policy.

Synopsis

```
set authentication Policy <name> [-rule <expression>] [-action <string>] [-undefAction <string>] [-comment <string>] [-logAction <string>]
```

Arguments

name

Name of the advance authentication policy to modify.

rule

Name of the NetScaler named rule, or a default syntax expression, that the policy uses to determine whether to attempt to authenticate the user with the AUTHENTICATION server.

action

Name of the authentication action to be performed if the policy matches.

undefAction

Action to perform if the result of policy evaluation is undefined (UNDEF). An UNDEF event indicates an internal error condition. Only the above built-in actions can be used.

comment

Any comments to preserve information about this policy.

logAction

Name of messagelog action to use when a request matches this policy.

unset authentication Policy

Use this command to remove authentication Policy settings. Refer to the set authentication Policy command for meanings of the arguments.

Synopsis

```
unset authentication Policy <name> [-undefAction] [-comment] [-logAction]
```

show authentication Policy

Displays the current settings for the specified advance authentication policy. If no policy name is provided, displays a list of all advance authentication policies currently configured on the NetScaler appliance.

Synopsis

```
show authentication Policy [<name>]
```

Arguments

name

Name of the advance authentication policy.

Outputs

rule

The name of the new rule associated with the policy.

action

The name of the authentication action associated with the policy.

stateflag

undefAction

Action to perform if the result of policy evaluation is undefined (UNDEF). An UNDEF event indicates an internal error condition. Only the above built-in actions can be used.

comment

Any comments to preserve information about this policy.

logAction

Name of message log action to use when a request matches this policy.

hits

Number of hits.

boundTo

The entity name to which policy is bound

activePolicy

priority

bindPolicyType

policyType

nextFactor

On success invoke label.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

description

Description of the policy

vserverType

policySubType

devno

count

rename authentication Policy

Renames the specified authentication policy.

Synopsis

rename authentication Policy <name>@ <newName>@

Arguments

name

Existing name of the authentication policy.

newName

New name for the authentication policy. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) hash (#), space (), at (@), equals (=), colon (:), and underscore characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my authentication policy" or 'my authentication policy').

Example

```
rename authentication policy oldname newname
```

stat authentication Policy

Displays authentication statistics for all advanced authentication policies, or for only the specified policy.

Synopsys

```
stat authentication Policy [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

name

Name of the advanced authentication policy for which to display statistics. If no name is specified, statistics for all advanced authentication policies are shown.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Policy hits (Hits)

Number of hits on the policy

Policy undef hits (Undefhits)

Number of undef hits on the policy

Example

```
stat authentication policy
```

authentication authnProfile

The following operations can be performed on "authentication authnProfile":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add authentication authnProfile

Creates an authentication profile to hold all authentication related configuration for TM vserver.

Synopsys

```
add authentication authnProfile <name> {-authnVsName <string>} {-AuthenticationHost <string>} {-AuthenticationDomain <string>} [-AuthenticationLevel <positive_integer>]
```

Arguments

name

Name for the authentication profile.

Must begin with a letter, number, or the underscore character (`_`), and must contain only letters, numbers, and the hyphen (`-`), period (`.`), pound (`#`), space (), at (`@`), equals (`=`), colon (`:`), and underscore characters. Cannot be changed after the RADIUS action is added.

authnVsName

Name of the authentication vserver at which authentication should be done.

Maximum value: 128

AuthenticationHost

Hostname of the authentication vserver.

Maximum value: 256

AuthenticationDomain

Domain for which TM cookie must to be set. If unspecified, cookie will be set for FQDN.

Maximum value: 256

AuthenticationLevel

Authentication weight or level of the vserver to which this will bound. This is used to order TM vservers based on the protection required. A session that is created by authenticating against TM vserver at given level cannot be used to access TM vserver at a higher level.

Minimum value: 0

Maximum value: 255

rm authentication authnProfile

Removes an authentication profile. A profile cannot be removed as long as it is set to a vserver.

Synopsys

```
rm authentication authnProfile <name>
```

Arguments

name

Name of the authentication profile to be removed.

set authentication authnProfile

Configures an authentication profile.

Synopsys

```
set authentication authnProfile <name> [-authnVsName <string>] [-AuthenticationHost <string>] [-AuthenticationDomain <string>] [-AuthenticationLevel <positive_integer>]
```

Arguments

name

Name of the authentication profile.

authnVsName

Name of the authentication vserver at which authentication should be done.

Maximum value: 128

AuthenticationHost

Hostname of the authentication vserver.

Maximum value: 256

AuthenticationDomain

Domain for which TM cookie must to be set. If unspecified, cookie will be set for FQDN.

Maximum value: 256

AuthenticationLevel

Authentication weight or level of the vserver to which this will bound. This is used to order TM vservers based on the protection required. A session that is created by authenticating against TM vserver at given level cannot be used to access TM vserver at a higher level.

Minimum value: 0

Maximum value: 255

unset authentication authnProfile

Use this command to remove authentication authnProfile settings. Refer to the set authentication authnProfile command for meanings of the arguments.

Synopsys

```
unset authentication authnProfile <name> [-AuthenticationDomain] [-AuthenticationLevel]
```

show authentication authnProfile

Displays the current configuration for the authentication profile specified

Synopsys

```
show authentication authnProfile [<name>]
```

Arguments

name

Name of the authentication profile.

Outputs

authnVsName

Name of the authentication vserver at which authentication should be done.

AuthenticationHost

Hostname of the authentication vserver.

AuthenticationDomain

Domain for which TM cookie must to be set. If unspecified, cookie will be set for FQDN.

AuthenticationLevel

Authentication weight or level of the vserver to which this will bound. This is used to order TM vservers based on the protection required. A session that is created by authenticating against TM vserver at given level cannot be used to access TM vserver at a higher level.

devno

count

stateflag

authentication certAction

The following operations can be performed on "authentication certAction":

add | **rm** | **set** | **unset** | **show**

add authentication certAction

Adds an action (profile) for a client certificate (cert) authentication server. The profile contains all configuration data necessary to communicate with that client cert authentication server.

Synopsis

```
add authentication certAction <name> [-twoFactor ( ON | OFF )] [-userNameField <string>] [-groupNameField <string>] [-defaultAuthenticationGroup <string>]
```

Arguments

name

Name for the client cert authentication server profile (action).

Must begin with a letter, number, or the underscore character (), and must contain only letters, numbers, and the hyphen (-), period (.), pound (#), space (), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after certificate action is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my authentication action? or ?my authentication action?).

twoFactor

Enables or disables two-factor authentication.

Two factor authentication is client cert authentication followed by password authentication.

Possible values: ON, OFF

Default value: OFF

userNameField

Client-cert field from which the username is extracted. Must be set to either ""Subject"" and ""Issuer"" (include both sets of double quotation marks).

Format: <field>:<subfield>.

groupNameField

Client-cert field from which the group is extracted. Must be set to either ""Subject"" and ""Issuer"" (include both sets of double quotation marks).

Format: <field>:<subfield>

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Maximum value: 64

Example

```
add authentication certaction -twoFactor ON -userNameField "Subject:CN" -groupNameField "
```

rm authentication certAction

Removes an existing client cert authentication server profile (action).

Synopsys

```
rm authentication certAction <name>
```

Arguments

name

Name of the profile to be removed.

set authentication certAction

Configures a client cert authentication server profile (action).

Synopsys

```
set authentication certAction <name> [-twoFactor ( ON | OFF )] [-userNameField <string>] [-groupNameField <string>] [-defaultAuthenticationGroup <string>]
```

Arguments

name

Name of the client cert server profile.

twoFactor

Enables or disables two-factor authentication.

Two factor authentication is client cert authentication followed by password authentication.

Possible values: ON, OFF

Default value: OFF

userNameField

Client-cert field from which the username is extracted. Must be set to either ""Subject"" and ""Issuer"" (include both sets of double quotation marks).

Format: <field>:<subfield>.

groupNameField

Client-cert field from which the group is extracted. Must be set to either ""Subject"" and ""Issuer"" (include both sets of double quotation marks).

Format: <field>:<subfield>

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Maximum value: 64

Example

```
set authentication certaction -twoFactor ON -userNameField "Subject:CN" -groupNameField ";
```

unset authentication certAction

Use this command to remove authentication certAction settings. Refer to the set authentication certAction command for meanings of the arguments.

Synopsys

unset authentication certAction <name> [-twoFactor] [-userNameField] [-groupNameField] [-defaultAuthenticationGroup]

show authentication certAction

Displays the current configuration settings for the specified client cert authentication server profile (action).

Synopsys

show authentication certAction [<name>]

Arguments

name

Name of the client cert server profile (action).

Outputs

twoFactor

The state of two factor authentication.

userNameField

The field in the certificate from which the username will be extracted.

groupNameField

The field in the certificate from which the group will be extracted.

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

stateflag

devno

count

authentication certPolicy

The following operations can be performed on "authentication certPolicy":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add authentication certPolicy

Adds a client certificate (cert) authentication policy. The policy defines the criteria under which the NetScaler appliance attempts to authenticate the user with the specified client cert authentication server.

Synopsys

add authentication certPolicy <name> <rule> [<reqAction>]

Arguments

name

Name for the client certificate authentication policy.

Must begin with a letter, number, or the underscore character (`_`), and must contain only letters, numbers, and the hyphen (`-`), period (`.`) pound (`#`), space (), at (`@`), equals (`=`), colon (`:`), and underscore characters. Cannot be changed after cert authentication policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, `?my authentication policy?` or `'my authentication policy?'`).

rule

Name of the NetScaler named rule, or a default syntax expression, that the policy uses to determine whether to attempt to authenticate the user with the authentication server.

reqAction

Name of the client cert authentication action to be performed if the policy matches.

rm authentication certPolicy

Removes a client cert authentication policy.

Synopsys

rm authentication certPolicy <name>

Arguments

name

Name of the client cert policy to remove.

set authentication certPolicy

Configures the specified client cert authentication policy.

Synopsys

set authentication certPolicy <name> [-rule <expression>] [-reqAction <string>]

Arguments

name

Name of the client cert policy.

rule

Name of the NetScaler named rule, or a default syntax expression, that the policy uses to determine whether to attempt to authenticate the user with the authentication server.

reqAction

Name of the client cert authentication action to be performed if the policy matches.

unset authentication certPolicy

Use this command to remove authentication certPolicy settings. Refer to the set authentication certPolicy command for meanings of the arguments.

Synopsys

unset authentication certPolicy <name> [-rule] [-reqAction]

show authentication certPolicy

Displays the current settings for the specified client cert authentication policy. If no policy name is provided, displays a list of all client cert authentication policies currently configured on the NetScaler appliance.

Synopsys

show authentication certPolicy [<name>]

Arguments

name

Name of the client cert authentication policy.

Outputs

rule

The rule associated with the policy.

reqAction

The cert action associated with the policy.

boundTo

The entity name to which policy is bound

activePolicy

priority

bindPolicyType

policyType

devno

count

stateflag

authentication ldapAction

The following operations can be performed on "authentication ldapAction":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add authentication ldapAction

Creates an action (profile) for an LDAP server. This profile contains all configuration data needed to communicate with that LDAP server.

Synopsys

```
add authentication ldapAction <name> {-serverIP <ip_addr|ipv6_addr|*> | {-serverName <string>}} [-serverPort <port>] [-authTimeout <positive_integer>] [-ldapBase <string>] [-ldapBindDn <string>] {-ldapBindDnPassword } [-ldapLoginName <string>] [-searchFilter <string>] [-groupAttrName <string>] [-subAttributeName <string>] [-secType <secType>] [-svrType ( AD | NDS )] [-ssoNameAttribute <string>] [-authentication ( ENABLED | DISABLED )] [-requireUser ( YES | NO )] [-passwdChange ( ENABLED | DISABLED )] [-nestedGroupExtraction ( ON | OFF )] [-maxNestingLevel <positive_integer>] [-groupSearchSubAttribute <string>] [-groupSearchFilter <string>]] [-followReferrals ( ON | OFF )] [-maxLDAPReferrals <positive_integer>]] [-validateServerCert ( YES | NO )] [-ldapHostname <string>] [-groupNameIdentifier <string>] [-groupSearchAttribute <string>] [-defaultAuthenticationGroup <string>] [-Attribute1 <string>] [-Attribute2 <string>] [-Attribute3 <string>] [-Attribute4 <string>] [-Attribute5 <string>] [-Attribute6 <string>] [-Attribute7 <string>] [-Attribute8 <string>] [-Attribute9 <string>] [-Attribute10 <string>] [-Attribute11 <string>] [-Attribute12 <string>] [-Attribute13 <string>] [-Attribute14 <string>] [-Attribute15 <string>] [-Attribute16 <string>]
```

Arguments

name

Name for the new LDAP action.

Must begin with a letter, number, or the underscore character (), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space (), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after the LDAP action is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my authentication action? or ?my authentication action?).

serverIP

IP address assigned to the LDAP server.

serverName

LDAP server name as a FQDN. Mutually exclusive with LDAP IP address.

serverPort

Port on which the LDAP server accepts connections.

Default value: 389

Minimum value: 1

authTimeout

Number of seconds the NetScaler appliance waits for a response from the RADIUS server.

Default value: 3

Minimum value: 1

ldapBase

Base (node) from which to start LDAP searches.

If the LDAP server is running locally, the default value of base is dc=netcaler, dc=com.

IdapBindDn

Full distinguished name (DN) that is used to bind to the LDAP server.

Default: cn=Manager,dc=netcaler,dc=com

IdapBindDnPassword

Password used to bind to the LDAP server.

IdapLoginName

LDAP login name attribute.

The NetScaler appliance uses the LDAP login name to query external LDAP servers or Active Directories.

searchFilter

String to be combined with the default LDAP user search string to form the search value. For example, if the search filter "?vpnallowed=true"? is combined with the LDAP login name "?samaccount"? and the user-supplied username is "?bob"? , the result is the LDAP search string "(&(vpnallowed=true)(samaccount=bob)" (Be sure to enclose the search string in two sets of double quotation marks; both sets are needed.).

groupAttrName

LDAP group attribute name.

Used for group extraction on the LDAP server.

subAttributeName

LDAP group sub-attribute name.

Used for group extraction from the LDAP server.

secType

Type of security used for communications between the NetScaler appliance and the LDAP server. For the PLAINTEXT setting, no encryption is required.

Possible values: PLAINTEXT, TLS, SSL

Default value: PLAINTEXT

svrType

The type of LDAP server.

Possible values: AD, NDS

Default value: AAA_LDAP_SERVER_TYPE_DEFAULT

ssoNameAttribute

LDAP single signon (SSO) attribute.

The NetScaler appliance uses the SSO name attribute to query external LDAP servers or Active Directories for an alternate username.

authentication

Perform LDAP authentication.

If authentication is disabled, any LDAP authentication attempt returns authentication success if the user is found.

CAUTION! Authentication should be disabled only for authorization group extraction or where other (non-LDAP) authentication methods are in use and either bound to a primary list or flagged as secondary.

Possible values: ENABLED, DISABLED

Default value: ENABLED

requireUser

Require a successful user search for authentication.

Possible values: YES, NO

Default value: YES

passwdChange

Allow password change requests.

Possible values: ENABLED, DISABLED

Default value: DISABLED

nestedGroupExtraction

Allow nested group extraction, in which the NetScaler appliance queries external LDAP servers to determine whether a group is part of another group.

Possible values: ON, OFF

Default value: OFF

maxNestingLevel

If nested group extraction is ON, specifies the number of levels up to which group extraction is performed.

Default value: 2

Minimum value: 2

followReferrals

Setting this option to ON enables following LDAP referrals received from the LDAP server.

Possible values: ON, OFF

Default value: OFF

maxLDAPReferrals

Specifies the maximum number of nested referrals to follow.

Default value: 1

Minimum value: 1

validateServerCert

When to validate LDAP server certs

Possible values: YES, NO

Default value: NO

ldapHostname

Hostname for the LDAP server. If -validateServerCert is ON then this must be the host name on the certificate from the LDAP server.

A hostname mismatch will cause a connection failure.

groupNameIdentifier

Name that uniquely identifies a group in LDAP or Active Directory.

groupSearchAttribute

LDAP group search attribute.

Used to determine to which groups a group belongs.

groupSearchSubAttribute

LDAP group search subattribute.

Used to determine to which groups a group belongs.

groupSearchFilter

String to be combined with the default LDAP group search string to form the search value. For example, the group search filter `"vpnallowed=true"` when combined with the group identifier `"samaccount"` and the group name `"g1"` yields the LDAP search string `"(&(vpnallowed=true)(samaccount=g1))"`. (Be sure to enclose the search string in two sets of double quotation marks; both sets are needed.)

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Maximum value: 64

Attribute1

Expression that would be evaluated to extract attribute1 from the ldap response

Maximum value: 64

Attribute2

Expression that would be evaluated to extract attribute2 from the ldap response

Maximum value: 64

Attribute3

Expression that would be evaluated to extract attribute3 from the ldap response

Maximum value: 64

Attribute4

Expression that would be evaluated to extract attribute4 from the ldap response

Maximum value: 64

Attribute5

Expression that would be evaluated to extract attribute5 from the ldap response

Maximum value: 64

Attribute6

Expression that would be evaluated to extract attribute6 from the ldap response

Maximum value: 64

Attribute7

Expression that would be evaluated to extract attribute7 from the ldap response

Maximum value: 64

Attribute8

Expression that would be evaluated to extract attribute8 from the ldap response

Maximum value: 64

Attribute9

Expression that would be evaluated to extract attribute9 from the ldap response

Maximum value: 64

Attribute10

Expression that would be evaluated to extract attribute10 from the ldap response

Maximum value: 64

Attribute11

Expression that would be evaluated to extract attribute11 from the ldap response

Maximum value: 64

Attribute12

Expression that would be evaluated to extract attribute12 from the ldap response

Maximum value: 64

Attribute13

Expression that would be evaluated to extract attribute13 from the ldap response

Maximum value: 64

Attribute14

Expression that would be evaluated to extract attribute14 from the ldap response

Maximum value: 64

Attribute15

Expression that would be evaluated to extract attribute15 from the ldap response

Maximum value: 64

Attribute16

Expression that would be evaluated to extract attribute16 from the ldap response

Maximum value: 64

rm authentication ldapAction

Removes an LDAP profile (action). NOTE: An action cannot be removed if it is bound to a policy.

Synopsis

rm authentication ldapAction <name>

Arguments

name

Name of the LDAP profile (action) to be removed.

set authentication ldapAction

Modifies an LDAP server profile (action.) The profile contains all configuration data needed to communicate with that LDAP server.

Synopsys

```
set authentication ldapAction <name> [-serverIP <ip_addr|ipv6_addr|*>] [-serverName <string>] [-serverPort <port>]
[-authTimeout <positive_integer>] [-ldapBase <string>] [-ldapBindDn <string>] {-ldapBindDnPassword } [-
ldapLoginName <string>] [-searchFilter <string>] [-groupAttrName <string>] [-subAttributeName <string>] [-secType
<secType>] [-svrType ( AD | NDS )] [-ssoNameAttribute <string>] [-authentication ( ENABLED | DISABLED )] [-
requireUser ( YES | NO )] [-passwdChange ( ENABLED | DISABLED )] [-validateServerCert ( YES | NO )] [-
ldapHostname <string>] [-nestedGroupExtraction ( ON | OFF )] [-maxNestingLevel <positive_integer>] [-
groupNameIdentifier <string>] [-groupSearchAttribute <string>] [-groupSearchSubAttribute <string>]] [-
groupSearchFilter <string>] [-followReferrals ( ON | OFF )] [-maxLDAPReferrals <positive_integer>] [-
defaultAuthenticationGroup <string>] [-Attribute1 <string>] [-Attribute2 <string>] [-Attribute3 <string>] [-Attribute4
<string>] [-Attribute5 <string>] [-Attribute6 <string>] [-Attribute7 <string>] [-Attribute8 <string>] [-Attribute9 <string>] [-
Attribute10 <string>] [-Attribute11 <string>] [-Attribute12 <string>] [-Attribute13 <string>] [-Attribute14 <string>] [-
Attribute15 <string>] [-Attribute16 <string>]
```

Arguments

name

Name of the LDAP profile to modify.

serverIP

IP address assigned to the LDAP server.

serverName

LDAP server name as a FQDN. Mutually exclusive with LDAP IP address.

serverPort

Port on which the LDAP server accepts connections.

Default value: 389

Minimum value: 1

authTimeout

Number of seconds the NetScaler appliance waits for a response from the RADIUS server.

Default value: 3

Minimum value: 1

ldapBase

Base (node) from which to start LDAP searches.

If the LDAP server is running locally, the default value of base is dc=netScaler, dc=com.

ldapBindDn

Full distinguished name (DN) that is used to bind to the LDAP server.

Default: cn=Manager,dc=netScaler,dc=com

ldapBindDnPassword

Password used to bind to the LDAP server.

ldapLoginName

LDAP login name attribute.

The NetScaler appliance uses the LDAP login name to query external LDAP servers or Active Directories.

searchFilter

String to be combined with the default LDAP user search string to form the search value. For example, if the search filter `"?vpnallowed=true?"` is combined with the LDAP login name `"samaccount?"` and the user-supplied username is `"bob?"`, the result is the LDAP search string `"(&(vpnallowed=true)(samaccount=bob))"` (Be sure to enclose the search string in two sets of double quotation marks; both sets are needed.).

groupAttrName

LDAP group attribute name.

Used for group extraction on the LDAP server.

subAttributeName

LDAP group sub-attribute name.

Used for group extraction from the LDAP server.

secType

Type of security used for communications between the NetScaler appliance and the LDAP server. For the PLAINTEXT setting, no encryption is required.

Possible values: PLAINTEXT, TLS, SSL

Default value: PLAINTEXT

svrType

The type of LDAP server.

Possible values: AD, NDS

Default value: AAA_LDAP_SERVER_TYPE_DEFAULT

ssoNameAttribute

LDAP single signon (SSO) attribute.

The NetScaler appliance uses the SSO name attribute to query external LDAP servers or Active Directories for an alternate username.

authentication

Perform LDAP authentication.

If authentication is disabled, any LDAP authentication attempt returns authentication success if the user is found.

CAUTION! Authentication should be disabled only for authorization group extraction or where other (non-LDAP) authentication methods are in use and either bound to a primary list or flagged as secondary.

Possible values: ENABLED, DISABLED

Default value: ENABLED

requireUser

Require a successful user search for authentication.

Possible values: YES, NO

Default value: YES

passwdChange

Allow password change requests.

Possible values: ENABLED, DISABLED

Default value: DISABLED

validateServerCert

When to validate LDAP server certs

Possible values: YES, NO

Default value: NO

ldapHostname

Hostname for the LDAP server. If -validateServerCert is ON then this must be the host name on the certificate from the LDAP server.

A hostname mismatch will cause a connection failure.

nestedGroupExtraction

Allow nested group extraction, in which the NetScaler appliance queries external LDAP servers to determine whether a group is part of another group.

Possible values: ON, OFF

Default value: OFF

maxNestingLevel

If nested group extraction is ON, specifies the number of levels up to which group extraction is performed.

Default value: 2

Minimum value: 2

groupNameIdentifier

Name that uniquely identifies a group in LDAP or Active Directory.

groupSearchAttribute

LDAP group search attribute.

Used to determine to which groups a group belongs.

groupSearchSubAttribute

LDAP group search subattribute.

Used to determine to which groups a group belongs.

groupSearchFilter

String to be combined with the default LDAP group search string to form the search value. For example, the group search filter "vpnallowed=true" when combined with the group identifier "samaccount" and the group name "g1" yields the LDAP search string "(&(vpnallowed=true)(samaccount=g1)". (Be sure to enclose the search string in two sets of double quotation marks; both sets are needed.)

followReferrals

Setting this option to ON enables following LDAP referrals received from the LDAP server.

Possible values: ON, OFF

Default value: OFF

maxLDAPReferrals

Specifies the maximum number of nested referrals to follow.

Default value: 1

Minimum value: 1

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Maximum value: 64

Attribute1

Expression that would be evaluated to extract attribute1 from the ldap response

Maximum value: 64

Attribute2

Expression that would be evaluated to extract attribute2 from the ldap response

Maximum value: 64

Attribute3

Expression that would be evaluated to extract attribute3 from the ldap response

Maximum value: 64

Attribute4

Expression that would be evaluated to extract attribute4 from the ldap response

Maximum value: 64

Attribute5

Expression that would be evaluated to extract attribute5 from the ldap response

Maximum value: 64

Attribute6

Expression that would be evaluated to extract attribute6 from the ldap response

Maximum value: 64

Attribute7

Expression that would be evaluated to extract attribute7 from the ldap response

Maximum value: 64

Attribute8

Expression that would be evaluated to extract attribute8 from the ldap response

Maximum value: 64

Attribute9

Expression that would be evaluated to extract attribute9 from the ldap response

Maximum value: 64

Attribute10

Expression that would be evaluated to extract attribute10 from the ldap response

Maximum value: 64

Attribute11

Expression that would be evaluated to extract attribute11 from the ldap response

Maximum value: 64

Attribute12

Expression that would be evaluated to extract attribute12 from the ldap response

Maximum value: 64

Attribute13

Expression that would be evaluated to extract attribute13 from the ldap response

Maximum value: 64

Attribute14

Expression that would be evaluated to extract attribute14 from the ldap response

Maximum value: 64

Attribute15

Expression that would be evaluated to extract attribute15 from the ldap response

Maximum value: 64

Attribute16

Expression that would be evaluated to extract attribute16 from the ldap response

Maximum value: 64

unset authentication ldapAction

Use this command to remove authentication ldapAction settings. Refer to the set authentication ldapAction command for meanings of the arguments.

Synopsis

```
unset authentication ldapAction <name> [-serverIP] [-serverName] [-serverPort] [-authTimeout] [-ldapBase] [-ldapBindDn] [-ldapBindDnPassword] [-ldapLoginName] [-searchFilter] [-groupAttrName] [-subAttributeName] [-secType] [-svrType] [-ssoNameAttribute] [-authentication] [-requireUser] [-passwdChange] [-validateServerCert] [-ldapHostname] [-nestedGroupExtraction] [-maxNestingLevel] [-groupNameIdentifier] [-groupSearchAttribute] [-groupSearchSubAttribute] [-groupSearchFilter] [-followReferrals] [-maxLDAPReferrals] [-defaultAuthenticationGroup] [-Attribute1] [-Attribute2] [-Attribute3] [-Attribute4] [-Attribute5] [-Attribute6] [-Attribute7] [-Attribute8] [-Attribute9] [-Attribute10] [-Attribute11] [-Attribute12] [-Attribute13] [-Attribute14] [-Attribute15] [-Attribute16]
```

show authentication ldapAction

Displays the current configuration settings for the specified LDAP profile (action).

Synopsis

```
show authentication ldapAction [<name>]
```

Arguments

name

Name of the LDAP profile.

Outputs

serverIP

IP address assigned to the LDAP server.

serverName

LDAP server name as a FQDN. Mutually exclusive with LDAP IP address.

serverPort

Port on which the LDAP server accepts connections.

authTimeout

Number of seconds the NetScaler appliance waits for a response from the RADIUS server.

ldapBindDn

Full distinguished name (DN) that is used to bind to the LDAP server.

Default: cn=Manager,dc=netscaler,dc=com

ldapBindDnPassword

Password used to bind to the LDAP server.

ldapLoginName

LDAP login name attribute.

The NetScaler appliance uses the LDAP login name to query external LDAP servers or Active Directories.

ldapBase

Base (node) from which to start LDAP searches.

If the LDAP server is running locally, the default value of base is dc=netscaler, dc=com.

searchFilter

String to be combined with the default LDAP user search string to form the search value. For example, if the search filter "?vpnallowed=true"? is combined with the LDAP login name "?samaccount"? and the user-supplied username is "?bob"? , the result is the LDAP search string "(&(vpnallowed=true)(samaccount=bob))" (Be sure to enclose the search string in two sets of double quotation marks; both sets are needed.).

groupAttrName

LDAP group attribute name.

Used for group extraction on the LDAP server.

subAttributeName

LDAP group sub-attribute name.

Used for group extraction from the LDAP server.

secType

Type of security used for communications between the NetScaler appliance and the LDAP server. For the PLAINTEXT setting, no encryption is required.

svrType

The type of LDAP server.

ssoNameAttribute

LDAP single signon (SSO) attribute.

The NetScaler appliance uses the SSO name attribute to query external LDAP servers or Active Directories for an alternate username.

authentication

Perform LDAP authentication.

If authentication is disabled, any LDAP authentication attempt returns authentication success if the user is found.

CAUTION! Authentication should be disabled only for authorization group extraction or where other (non-LDAP) authentication methods are in use and either bound to a primary list or flagged as secondary.

requireUser

Require a successful user search for authentication.

Success

Failure

stateflag

nestedGroupExtraction

Allow nested group extraction, in which the NetScaler appliance queries external LDAP servers to determine whether a group is part of another group.

maxNestingLevel

If nested group extraction is ON, specifies the number of levels up to which group extraction is performed.

followReferrals

Setting this option to ON enables following LDAP referrals received from the LDAP server.

maxLDAPReferrals

Specifies the maximum number of nested referrals to follow.

validateServerCert

When to validate LDAP server certs

ldapHostname

Hostname for the LDAP server. If -validateServerCert is ON then this must be the host name on the certificate from the LDAP server.

A hostname mismatch will cause a connection failure.

groupNameIdentifier

Name that uniquely identifies a group in LDAP or Active Directory.

groupSearchAttribute

LDAP group search attribute.

Used to determine to which groups a group belongs.

groupSearchSubAttribute

LDAP group search subattribute.

Used to determine to which groups a group belongs.

groupSearchFilter

String to be combined with the default LDAP group search string to form the search value. For example, the group search filter ""vpnallowed=true"" when combined with the group identifier ""samaccount"" and the group name ""g1"" yields the LDAP search string ""(&(vpnallowed=true)(samaccount=g1)"". (Be sure to enclose the search string in two sets of double quotation marks; both sets are needed.)

passwdChange

Allow password change requests.

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Attribute1

Expression that would be evaluated to extract attribute1 from the ldap response

Attribute2

Expression that would be evaluated to extract attribute2 from the ldap response

Attribute3

Expression that would be evaluated to extract attribute3 from the ldap response

Attribute4

Expression that would be evaluated to extract attribute4 from the ldap response

Attribute5

Expression that would be evaluated to extract attribute5 from the ldap response

Attribute6

Expression that would be evaluated to extract attribute6 from the ldap response

Attribute7

Expression that would be evaluated to extract attribute7 from the ldap response

Attribute8

Expression that would be evaluated to extract attribute8 from the ldap response

Attribute9

Expression that would be evaluated to extract attribute9 from the ldap response

Attribute10

Expression that would be evaluated to extract attribute10 from the ldap response

Attribute11

Expression that would be evaluated to extract attribute11 from the ldap response

Attribute12

Expression that would be evaluated to extract attribute12 from the ldap response

Attribute13

Expression that would be evaluated to extract attribute13 from the ldap response

Attribute14

Expression that would be evaluated to extract attribute14 from the ldap response

Attribute15

Expression that would be evaluated to extract attribute15 from the ldap response

Attribute16

Expression that would be evaluated to extract attribute16 from the ldap response

devno

count

authentication ldapPolicy

The following operations can be performed on "authentication ldapPolicy":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add authentication ldapPolicy

Adds an LDAP authentication policy. The policy defines the criteria under which the NetScaler appliance attempts to authenticate the user with the specified LDAP server.

Synopsis

```
add authentication ldapPolicy <name> <rule> [<reqAction>]
```

Arguments

name

Name for the LDAP policy.

Must begin with a letter, number, or the underscore character (`_`), and must contain only letters, numbers, and the hyphen (`-`), period (`.`), pound (`#`), space (), at (`@`), equals (`=`), colon (`:`), and underscore characters. Cannot be changed after LDAP policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, `?my authentication policy?` or `'my authentication policy?'`).

rule

Name of the NetScaler named rule, or a default syntax expression, that the policy uses to determine whether to attempt to authenticate the user with the LDAP server.

reqAction

Name of the LDAP action to perform if the policy matches.

rm authentication ldapPolicy

Removes an LDAP policy.

Synopsis

```
rm authentication ldapPolicy <name>
```

Arguments

name

Name of the LDAP policy to remove.

set authentication ldapPolicy

Configures the specified LDAP policy.

Synopsis

```
set authentication ldapPolicy <name> [-rule <string>] [-reqAction <string>]
```

Arguments

name

Name of the LDAP policy.

rule

The new rule to associate with the policy.

reqAction

The new LDAP action to associate with the policy.

unset authentication IdapPolicy

Use this command to remove authentication IdapPolicy settings. Refer to the set authentication IdapPolicy command for meanings of the arguments.

Synopsys

unset authentication IdapPolicy <name> [-rule] [-reqAction]

show authentication IdapPolicy

Displays the current settings for the specified LDAP policy. If no policy name is provided, displays a list of all LDAP policies currently configured on the NetScaler appliance.

Synopsys

show authentication IdapPolicy [<name>]

Arguments

name

Name of the LDAP policy.

Outputs

rule

Name of the NetScaler named rule, or a default syntax expression, that the policy uses to determine whether to attempt to authenticate the user with the LDAP server.

reqAction

Name of the LDAP action to perform if the policy matches.

boundTo

The entity name to which policy is bound

activePolicy

priority

bindPolicyType

policyType

devno

count

stateflag

authentication localPolicy

The following operations can be performed on "authentication localPolicy":

[add](#) | [rm](#) | [set](#) | [show](#)

add authentication localPolicy

Adds a policy for the NetScaler appliance to locally authenticate a user. The policy contains criteria that specify when and how to authenticate a user.

Synopsys

add authentication localPolicy <name> <rule>

Arguments

name

Name for the local authentication policy.

Must begin with a letter, number, or the underscore character (`_`), and must contain only letters, numbers, and the hyphen (`-`), period (`.`) pound (`#`), space (), at (`@`), equals (`=`), colon (`:`), and underscore characters. Cannot be changed after local policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, `?my authentication policy?` or `'my authentication policy?'`).

rule

Name of the NetScaler named rule, or a default syntax expression, that the policy uses to perform the authentication.

rm authentication localPolicy

Removes the specified local authentication policy.

Synopsys

rm authentication localPolicy <name>

Arguments

name

Name of the local policy to remove.

set authentication localPolicy

Configures the specified local authentication policy.

Synopsys

set authentication localPolicy <name> -rule <expression>

Arguments

name

Name of the local authentication policy.

rule

Name of the NetScaler named rule, or a default syntax expression, that the policy uses to perform the authentication.

show authentication localPolicy

Displays the current settings for the specified local authentication policy. If no policy name is provided, displays a list of all local authentication policies currently configured on the NetScaler appliance.

Synopsys

show authentication localPolicy [<name>]

Arguments

name

Name of the local authentication policy.

Outputs

rule

The new rule associated with the policy.

boundTo

The entity name to which policy is bound

activePolicy

priority

reqAction

The name of the RADIUS action the policy uses

bindPolicyType

policyType

devno

count

stateflag

authentication negotiateAction

The following operations can be performed on "authentication negotiateAction":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add authentication negotiateAction

Creates an action (profile) for an Active Directory (AD) server that is used as a Kerberos Key Distribution Center (KDC). The profile contains all configuration data necessary to communicate with that AD KDC server.

Synopsis

```
add authentication negotiateAction <name> {-domain <string>} {-domainUser <string>} {-domainUserPasswd } [-defaultAuthenticationGroup <string>] [-keytab <string>]
```

Arguments

name

Name for the AD KDC server profile (negotiate action).

Must begin with a letter, number, or the underscore character (`_`), and must contain only letters, numbers, and the hyphen (`-`), period (`.`) pound (`#`), space (), at (`@`), equals (`=`), colon (`:`), and underscore characters. Cannot be changed after AD KDC server profile is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, `?my authentication action?` or `'my authentication action?'`).

domain

Domain name of the AD KDC server.

domainUser

User name that the NetScaler appliance uses to join the AD KDC server domain.

The NetScaler appliance uses the domain user name to check the health of the AD KDC server.

domainUserPasswd

Password that the NetScaler appliance uses to join the AD KDC server domain.

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Maximum value: 64

keytab

The path to the keytab file

rm authentication negotiateAction

Removes an AD KDC server profile (negotiate action). An action cannot be removed if it is bound to a policy.

Synopsis

```
rm authentication negotiateAction <name>
```

Arguments

name

Name of the AD KDC server profile to be removed.

set authentication negotiateAction

Configures an AD KDC server profile (negotiate action).

Synopsis

```
set authentication negotiateAction <name> [-domain <string>] [-domainUser <string>] [-domainUserPasswd ] [-defaultAuthenticationGroup <string>] [-keytab <string>]
```

Arguments

name

Name of the AD KDC server profile.

domain

Domain name of the AD KDC server.

domainUser

User name that the NetScaler appliance uses to join the AD KDC server domain.

The NetScaler appliance uses the domain user name to check the health of the AD KDC server.

domainUserPasswd

Password that the NetScaler appliance uses to join the AD KDC server domain.

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Maximum value: 64

keytab

The path to the keytab file

unset authentication negotiateAction

Use this command to remove authentication negotiateAction settings. Refer to the set authentication negotiateAction command for meanings of the arguments.

Synopsis

```
unset authentication negotiateAction <name> [-domain] [-domainUser] [-domainUserPasswd] [-defaultAuthenticationGroup]
```

show authentication negotiateAction

Displays the current configuration settings for the specified AD KDC server profile (negotiate action).

Synopsis

```
show authentication negotiateAction [<name>]
```

Arguments

name

Name of the AD KDC server profile.

Outputs

domain

Domain name of the AD KDC server.

domainUser

User name that the NetScaler appliance uses to join the AD KDC server domain.

The NetScaler appliance uses the domain user name to check the health of the AD KDC server.

domainUserPasswd

Password that the NetScaler appliance uses to join the AD KDC server domain.

OU

Active Directory organizational units (OU) attribute.

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

keytab

The path to the keytab file

kcdSPN

Host SPN extracted from keytab file.

stateflag

devno

count

authentication negotiatePolicy

The following operations can be performed on "authentication negotiatePolicy":

[add](#) | [rm](#) | [set](#) | [show](#)

add authentication negotiatePolicy

Adds an Active Directory (AD) Kerberos Key Distribution Center (KCD) authentication policy (negotiate policy). The policy defines the criteria under which the NetScaler appliance attempts to authenticate the user with the specified AD KCD server.

Synopsis

```
add authentication negotiatePolicy <name> <rule> <reqAction>
```

Arguments

name

Name for the negotiate authentication policy.

Must begin with a letter, number, or the underscore character (`_`), and must contain only letters, numbers, and the hyphen (`-`), period (`.`) pound (`#`), space (), at (`@`), equals (`=`), colon (`:`), and underscore characters. Cannot be changed after AD KCD (negotiate) policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, `?my authentication policy?` or `'my authentication policy'`).

rule

Name of the NetScaler named rule, or a default syntax expression, that the policy uses to determine whether to attempt to authenticate the user with the AD KCD server.

reqAction

Name of the negotiate action to perform if the policy matches.

rm authentication negotiatePolicy

Removes the specified AD KCD (negotiate) policy.

Synopsis

```
rm authentication negotiatePolicy <name>
```

Arguments

name

Name of the negotiate policy to remove.

set authentication negotiatePolicy

Modifies the specified AD KCD (negotiate) policy.

Synopsis

```
set authentication negotiatePolicy <name> [-rule <expression>] [-reqAction <string>]
```

Arguments

name

Name of the negotiate policy to modify.

rule

Name of the NetScaler named rule, or a default syntax expression, that the policy uses to determine whether to attempt to authenticate the user with the AD KCD server.

reqAction

Name of the negotiate action to perform if the policy matches.

show authentication negotiatePolicy

Displays the current settings for the specified AD KCD (negotiate) policy. If no policy name is provided, displays a list of all negotiate policies currently configured on the NetScaler appliance.

Synopsys

show authentication negotiatePolicy [<name>]

Arguments

name

Name of the negotiate policy.

Outputs

rule

The name of the new rule associated with the policy.

reqAction

The name of the Negotiate action associated with the policy.

boundTo

The entity name to which policy is bound

activePolicy**priority****bindPolicyType****policyType****devno****count****stateflag**

authentication policylabel

The following operations can be performed on "authentication policylabel":

add | **rm** | **bind** | **unbind** | **rename** | **show** | **stat**

add authentication policylabel

Creates a user-defined authentication policy label.

Synopsys

add authentication policylabel <labelName>

Arguments

labelName

Name for the new authentication policy label.

Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.), pound (#), space (), at (@), equals (=), colon (:), and underscore characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my authentication policy label? or ?authentication policy label?).

Example

```
add authentication policylabel trans_http_url
```

rm authentication policylabel

Removes an authorization policy label.

Synopsys

rm authentication policylabel <labelName>

Arguments

labelName

Name of the authorization policy label to remove.

Example

```
rm authorization policylabel trans_http_url
```

bind authentication policylabel

Binds an authentication policy to <authentication policy label>.

Synopsys

bind authentication policylabel <labelName> -policyName <string> -priority <positive_integer> [-gotoPriorityExpression <expression>] [-nextFactor <string>]

Arguments

labelName

Name of the authentication policy label to which to bind the policy.

policyName

Name of the authentication policy to bind to the policy label.

priority

Positive integer specifying the priority of the policy. The lower the number, the higher the priority. Policies within a label are evaluated in the order of their priority numbers.

Minimum value: 1

Maximum value: 2147483647

gotoPriorityExpression

Expression or other value specifying the next policy to be evaluated if the current policy evaluates to TRUE. Specify one of the following values:

- * NEXT ? Evaluate the policy with the next higher priority number.
- * END ? End policy evaluation.
- * USE_INVOCATION_RESULT ? Applicable if this policy invokes another policy label. If the final goto in the invoked policy label has a value of END, the evaluation stops. If the final goto is anything other than END, the current policy label performs a NEXT.
- * A default syntax or classic expression that evaluates to a number.

If you specify an expression, the number to which it evaluates determines the next policy to evaluate, as follows:

- * If the expression evaluates to a higher numbered priority, the policy with that priority is evaluated next.
- * If the expression evaluates to the priority of the current policy, the policy with the next higher numbered priority is evaluated next.
- * If the expression evaluates to a number that is larger than the largest numbered priority, policy evaluation ends.

An UNDEF event is triggered if:

- * The expression is invalid.
- * The expression evaluates to a priority number that is smaller than the current policy's priority number.
- * The expression evaluates to a priority number that is between the current policy's priority number (say, 30) and the highest priority number (say, 100), but does not match any configured priority number (for example, the expression evaluates to the number 85). This example assumes that the priority number increments by 10 for every successive policy, and therefore a priority number of 85 does not exist in the policy label.

nextFactor

On success invoke label.

Example

```
i) bind authentication policylabel authn_label_1 -policyName authn_pol_1 -priority 1 ii
```

unbind authentication policylabel

Unbinds the specified policy from the specified authorization policy label.

Synopsys

```
unbind authentication policylabel <labelName> -policyName <string> [-priority <positive_integer>]
```

Arguments

labelName

Name for the new authentication policy label.

Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.), pound (#), space (), at (@), equals (=), colon (:), and underscore characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my authentication policy label? or ?authentication policy label?).

policyName

Name of the authentication policy to bind to the policy label.

priority

Priority of the NOPOLICY to be unbound.

Minimum value: 1

Maximum value: 2147483647

Example

```
unbind authorization policylabel trans_http_url pol_1
```

rename authentication policylabel

Rename a authn policy label.

Synopsys

```
rename authentication policylabel <labelName>@ <newName>@
```

Arguments

labelName

The name of the auth policy label

newName

The new name of the auth policy label

Example

```
rename authn policy label oldname newname
```

show authentication policylabel

Displays the current settings for the specified authentication policy label. If no policy name is provided, displays a list of all authentication policy labels currently configured on the NetScaler appliance.

Synopsys

```
show authentication policylabel [<labelName>]
```

Arguments

labelName

Name of the authorization policy label.

Outputs

stateflag

numpol

Number of policies bound to label.

hits

Number of times policy label was invoked.

policyName

Name of the authentication policy to bind to the policy label.

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

flowType

Flowtype of the bound authentication policy.

description

Description of the policylabel

flags

nextFactor

On success invoke label.

devno

count

Example

```
i) show authentication policylabel trans_http_url ii) show authentication policylabel
```

stat authentication policylabel

Displays statistics for the specified authentication policy label. If no authentication policy label is specified, displays a list of all authentication policy labels.

Synopsys

```
stat authentication policylabel [<labelName>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile  
<input_filename>] [-clearstats ( basic | full )]
```

Arguments

labelName

Name of the authentication policy label.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Policy Label Hits (Hits)

Number of times policy label was invoked.

authentication radiusAction

The following operations can be performed on "authentication radiusAction":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add authentication radiusAction

Creates an action (profile) for a RADIUS server. The profile contains all configuration data necessary to communicate with that RADIUS server.

Synopsys

```
add authentication radiusAction <name> {-serverIP <ip_addr|ipv6_addr|*> | {-serverName <string>}} [-serverPort <port>] [-authTimeout <positive_integer>] {-radKey } [-radNASip ( ENABLED | DISABLED )] [-radNASid <string>] [-radVendorID <positive_integer>] [-radAttributeType <positive_integer>] [-radGroupsPrefix <string>] [-radGroupSeparator <string>] [-passEncoding <passEncoding>] [-ipVendorID <positive_integer>] [-ipAttributeType <positive_integer>] [-accounting ( ON | OFF )] [-pwdVendorID <positive_integer>] [-pwdAttributeType <positive_integer>] [-defaultAuthenticationGroup <string>] [-callingstationid ( ENABLED | DISABLED )]
```

Arguments

name

Name for the RADIUS action.

Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space (), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after the RADIUS action is added.

serverIP

IP address assigned to the RADIUS server.

serverName

RADIUS server name as a FQDN. Mutually exclusive with RADIUS IP address.

serverPort

Port number on which the RADIUS server listens for connections.

Minimum value: 1

authTimeout

Number of seconds the NetScaler appliance waits for a response from the RADIUS server.

Default value: 3

Minimum value: 1

radKey

Key shared between the RADIUS server and the NetScaler appliance.

Required to allow the NetScaler appliance to communicate with the RADIUS server.

radNASip

If enabled, the NetScaler appliance IP address (NSIP) is sent to the RADIUS server as the Network Access Server IP (NASIP) address.

The RADIUS protocol defines the meaning and use of the NASIP address.

Possible values: ENABLED, DISABLED

radNASid

If configured, this string is sent to the RADIUS server as the Network Access Server ID (NASID).

radVendorID

RADIUS vendor ID attribute, used for RADIUS group extraction.

Minimum value: 1

radAttributeType

RADIUS attribute type, used for RADIUS group extraction.

Minimum value: 1

radGroupsPrefix

RADIUS groups prefix string.

This groups prefix precedes the group names within a RADIUS attribute for RADIUS group extraction.

radGroupSeparator

RADIUS group separator string

The group separator delimits group names within a RADIUS attribute for RADIUS group extraction.

passEncoding

Encoding type for passwords in RADIUS packets that the NetScaler appliance sends to the RADIUS server.

Possible values: pap, chap, mschapv1, mschapv2

Default value: pap

ipVendorID

Vendor ID of the intranet IP attribute in the RADIUS response.

NOTE: A value of 0 indicates that the attribute is not vendor encoded.

Minimum value: 0

ipAttributeType

Remote IP address attribute type in a RADIUS response.

Minimum value: 1

accounting

Whether the RADIUS server is currently accepting accounting messages.

Possible values: ON, OFF

pwdVendorID

Vendor ID of the attribute, in the RADIUS response, used to extract the user password.

Minimum value: 1

pwdAttributeType

Vendor-specific password attribute type in a RADIUS response.

Minimum value: 1

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Maximum value: 64

callingstationid

Send Calling-Station-ID of the client to the RADIUS server. IP Address of the client is sent as its Calling-Station-ID.

Possible values: ENABLED, DISABLED

Default value: DISABLED

rm authentication radiusAction

Removes a RADIUS profile (action). An action cannot be removed as long as it is bound to a policy.

Synopsys

```
rm authentication radiusAction <name>
```

Arguments

name

Name of the action to be removed.

set authentication radiusAction

Configures a RADIUS server profile (action). The profile contains all configuration data needed to communicate with that RADIUS server.

Synopsys

```
set authentication radiusAction <name> [-serverIP <ip_addr|ipv6_addr|*>] [-serverName <string>] [-serverPort <port>] [-authTimeout <positive_integer>] {-radKey } [-radNASip ( ENABLED | DISABLED )] [-radNASid <string>] [-radVendorID <positive_integer>] [-radAttributeType <positive_integer>] [-radGroupsPrefix <string>] [-radGroupSeparator <string>] [-passEncoding <passEncoding>] [-ipVendorID <positive_integer>] [-ipAttributeType <positive_integer>] [-accounting ( ON | OFF )] [-pwdVendorID <positive_integer>] [-pwdAttributeType <positive_integer>] [-defaultAuthenticationGroup <string>] [-callingstationid ( ENABLED | DISABLED )]
```

Arguments

name

Name of the RADIUS profile.

serverIP

IP address assigned to the RADIUS server.

serverName

RADIUS server name as a FQDN. Mutually exclusive with RADIUS IP address.

serverPort

Port number on which the RADIUS server listens for connections.

Minimum value: 1

authTimeout

Number of seconds the NetScaler appliance waits for a response from the RADIUS server.

Default value: 3

Minimum value: 1

radKey

Key shared between the RADIUS server and the NetScaler appliance.

Required to allow the NetScaler appliance to communicate with the RADIUS server.

radNASip

If enabled, the NetScaler appliance IP address (NSIP) is sent to the RADIUS server as the Network Access Server IP (NASIP) address.

The RADIUS protocol defines the meaning and use of the NASIP address.

Possible values: ENABLED, DISABLED

radNASid

If configured, this string is sent to the RADIUS server as the Network Access Server ID (NASID).

radVendorID

RADIUS vendor ID attribute, used for RADIUS group extraction.

Minimum value: 1

radAttributeType

RADIUS attribute type, used for RADIUS group extraction.

Minimum value: 1

radGroupsPrefix

RADIUS groups prefix string.

This groups prefix precedes the group names within a RADIUS attribute for RADIUS group extraction.

radGroupSeparator

RADIUS group separator string

The group separator delimits group names within a RADIUS attribute for RADIUS group extraction.

passEncoding

Encoding type for passwords in RADIUS packets that the NetScaler appliance sends to the RADIUS server.

Possible values: pap, chap, mschapv1, mschapv2

Default value: pap

ipVendorID

Vendor ID of the intranet IP attribute in the RADIUS response.

NOTE: A value of 0 indicates that the attribute is not vendor encoded.

Minimum value: 0

ipAttributeType

Remote IP address attribute type in a RADIUS response.

Minimum value: 1

accounting

Whether the RADIUS server is currently accepting accounting messages.

Possible values: ON, OFF

pwdVendorID

Vendor ID of the attribute, in the RADIUS response, used to extract the user password.

Minimum value: 1

pwdAttributeType

Vendor-specific password attribute type in a RADIUS response.

Minimum value: 1

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Maximum value: 64

callingstationid

Send Calling-Station-ID of the client to the RADIUS server. IP Address of the client is sent as its Calling-Station-ID.

Possible values: ENABLED, DISABLED

Default value: DISABLED

unset authentication radiusAction

Use this command to remove authentication radiusAction settings. Refer to the set authentication radiusAction command for meanings of the arguments.

Synopsis

```
unset authentication radiusAction <name> [-serverIP] [-serverName] [-serverPort] [-authTimeout] [-radNASip] [-radNASid] [-radVendorID] [-radAttributeType] [-radGroupsPrefix] [-radGroupSeparator] [-passEncoding] [-ipVendorID] [-ipAttributeType] [-accounting] [-pwdVendorID] [-pwdAttributeType] [-defaultAuthenticationGroup] [-callingstationid]
```

show authentication radiusAction

Displays the current configuration settings for the specified RADIUS profile (action).

Synopsis

```
show authentication radiusAction [<name>]
```

Arguments

name

Name of the RADIUS profile.

Outputs

serverIP

IP address assigned to the RADIUS server.

serverName

RADIUS server name as a FQDN. Mutually exclusive with RADIUS IP address.

serverPort

Port number on which the RADIUS server listens for connections.

authTimeout

Number of seconds the NetScaler appliance waits for a response from the RADIUS server.

radKey

Key shared between the RADIUS server and the NetScaler appliance.

Required to allow the NetScaler appliance to communicate with the RADIUS server.

radNASip

If enabled, the NetScaler appliance IP address (NSIP) is sent to the RADIUS server as the Network Access Server IP (NASIP) address.

The RADIUS protocol defines the meaning and use of the NASIP address.

IPAddress

IP address.

radNASid

If configured, this string is sent to the RADIUS server as the Network Access Server ID (NASID).

radVendorID

RADIUS vendor ID attribute, used for RADIUS group extraction.

radAttributeType

RADIUS attribute type, used for RADIUS group extraction.

radGroupsPrefix

RADIUS groups prefix string.

This groups prefix precedes the group names within a RADIUS attribute for RADIUS group extraction.

radGroupSeparator

RADIUS group separator string

The group separator delimits group names within a RADIUS attribute for RADIUS group extraction.

passEncoding

Encoding type for passwords in RADIUS packets that the NetScaler appliance sends to the RADIUS server.

ipVendorID

Vendor ID of the intranet IP attribute in the RADIUS response.

NOTE: A value of 0 indicates that the attribute is not vendor encoded.

ipAttributeType

Remote IP address attribute type in a RADIUS response.

accounting

Whether the RADIUS server is currently accepting accounting messages.

Success

Failure

stateflag

pwdVendorID

Vendor ID of the attribute, in the RADIUS response, used to extract the user password.

pwdAttributeType

Vendor-specific password attribute type in a RADIUS response.

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

callingstationid

Send Calling-Station-ID of the client to the RADIUS server. IP Address of the client is sent as its Calling-Station-ID.

devno**count**

authentication radiusPolicy

The following operations can be performed on "authentication radiusPolicy":

add | **rm** | **set** | **unset** | **show**

add authentication radiusPolicy

Adds a RADIUS authentication policy. The policy defines the criteria under which the NetScaler appliance attempts to authenticate the user with the RADIUS server.

Synopsis

add authentication radiusPolicy <name> <rule> [<reqAction>]

Arguments

name

Name for the RADIUS authentication policy.

Must begin with a letter, number, or the underscore character (`_`), and must contain only letters, numbers, and the hyphen (`-`), period (`.`), pound (`#`), space (), at (`@`), equals (`=`), colon (`:`), and underscore characters. Cannot be changed after RADIUS policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, `?my authentication policy?` or `'my authentication policy?'`).

rule

Name of the NetScaler named rule, or a default syntax expression, that the policy uses to determine whether to attempt to authenticate the user with the RADIUS server.

reqAction

Name of the RADIUS action to perform if the policy matches.

rm authentication radiusPolicy

Removes a RADIUS authentication policy.

Synopsis

rm authentication radiusPolicy <name>

Arguments

name

Name of the RADIUS authentication policy to remove.

set authentication radiusPolicy

Configures the specified RADIUS authentication policy.

Synopsis

set authentication radiusPolicy <name> [-rule <expression>] [-reqAction <string>]

Arguments

name

Name of the RADIUS authentication policy.

rule

Name of the NetScaler named rule, or a default syntax expression, that the policy uses to determine whether to attempt to authenticate the user with the RADIUS server.

reqAction

Name of the RADIUS action to perform if the policy matches.

unset authentication radiusPolicy

Use this command to remove authentication radiusPolicy settings. Refer to the set authentication radiusPolicy command for meanings of the arguments.

Synopsys

```
unset authentication radiusPolicy <name> [-rule] [-reqAction]
```

show authentication radiusPolicy

Displays the current settings for the specified RADIUS authentication policy. If no policy name is provided, displays a list of all RADIUS authentication policies currently configured on the NetScaler appliance.

Synopsys

```
show authentication radiusPolicy [<name>]
```

Arguments

name

Name of the RADIUS authentication policy.

Outputs

rule

The new rule associated with the policy.

reqAction

The new RADIUS action associated with the policy.

boundTo

The entity name to which policy is bound

activePolicy

priority

bindPolicyType

policyType

devno

count

stateflag

authentication samlAction

The following operations can be performed on "authentication samlAction":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add authentication samlAction

Creates an action (profile) for a Security Assertion Markup Language (SAML) server. The profile contains all configuration data necessary to communicate with that SAML server.

Synopsys

```
add authentication samlAction <name> {-samlIdPCertName <string>} {-samlSigningCertName <string>} {-samlRedirectUrl <string>} {-samlACSIndex <positive_integer>} {-samlUserField <string>} {-samlRejectUnsignedAssertion <samlRejectUnsignedAssertion>} {-samlIssuerName <string>} {-samlTwoFactor ( ON | OFF )} [-defaultAuthenticationGroup <string>] [-Attribute1 <string>] [-Attribute2 <string>] [-Attribute3 <string>] [-Attribute4 <string>] [-Attribute5 <string>] [-Attribute6 <string>] [-Attribute7 <string>] [-Attribute8 <string>] [-Attribute9 <string>] [-Attribute10 <string>] [-Attribute11 <string>] [-Attribute12 <string>] [-Attribute13 <string>] [-Attribute14 <string>] [-Attribute15 <string>] [-Attribute16 <string>] {-signatureAlg ( RSA-SHA1 | RSA-SHA256 )} {-digestMethod ( SHA1 | SHA256 )} [-requestedAuthnContext <requestedAuthnContext>] [-authnCtxClassRef <authnCtxClassRef> ...] [-samlBinding ( REDIRECT | POST )] [-attributeConsumingServiceIndex <positive_integer>] [-sendThumbprint ( ON | OFF )] [-logoutURL <string>]
```

Arguments

name

Name for the SAML server profile (action).

Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space (), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after SAML profile is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my authentication action? or ?my authentication action?).

samlIdPCertName

Name of the SAML server as given in that server's SSL certificate.

samlSigningCertName

Name of the signing authority as given in the SAML server's SSL certificate.

samlRedirectUrl

URL to which users are redirected for authentication.

samlACSIndex

Index/ID of the metadata entry corresponding to this configuration.

Default value: 255

Minimum value: 0

Maximum value: 255

samlUserField

SAML user ID, as given in the SAML assertion.

samlRejectUnsignedAssertion

Reject unsigned SAML assertions.

Possible values: ON, OFF, STRICT

Default value: ON

samlIssuerName

The name to be used in requests sent from Netscaler to IdP to uniquely identify Netscaler.

samlTwoFactor

Option to enable second factor after SAML

Possible values: ON, OFF

Default value: OFF

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Maximum value: 64

Attribute1

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute1

Maximum value: 64

Attribute2

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute2

Maximum value: 64

Attribute3

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute3

Maximum value: 64

Attribute4

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute4

Maximum value: 64

Attribute5

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute5

Maximum value: 64

Attribute6

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute6

Maximum value: 64

Attribute7

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute7

Maximum value: 64

Attribute8

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute8

Maximum value: 64

Attribute9

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute9

Maximum value: 64

Attribute10

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute10

Maximum value: 64

Attribute11

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute11

Maximum value: 64

Attribute12

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute12

Maximum value: 64

Attribute13

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute13

Maximum value: 64

Attribute14

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute14

Maximum value: 64

Attribute15

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute15

Maximum value: 64

Attribute16

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute16

Maximum value: 64

signatureAlg

Algorithm to be used to sign/verify SAML transactions

Possible values: RSA-SHA1, RSA-SHA256

Default value: RSA-SHA1

digestMethod

Algorithm to be used to compute/verify digest for SAML transactions

Possible values: SHA1, SHA256

Default value: SHA1

requestedAuthnContext

This element specifies the authentication context requirements of authentication statements returned in the response.

Possible values: exact, minimum, maximum, better

Default value: exact

authnCtxClassRef

This element specifies the authentication class types that are requested from IdP (IdentityProvider).

InternetProtocol: This is applicable when a principal is authenticated through the use of a provided IP address.

InternetProtocolPassword: This is applicable when a principal is authenticated through the use of a provided IP address, in addition to a username/password.

Kerberos: This is applicable when the principal has authenticated using a password to a local authentication authority, in order to acquire a Kerberos ticket.

MobileOneFactorUnregistered: This indicates authentication of the mobile device without requiring explicit end-user interaction.

MobileTwoFactorUnregistered: This indicates two-factor based authentication during mobile customer registration process, such as secure device and user PIN.

MobileOneFactorContract: Reflects mobile contract customer registration procedures and a single factor authentication.

MobileTwoFactorContract: Reflects mobile contract customer registration procedures and a two-factor based authentication.

Password: This class is applicable when a principal authenticates using password over unprotected http session.

PasswordProtectedTransport: This class is applicable when a principal authenticates to an authentication authority through the presentation of a password over a protected session.

PreviousSession: This class is applicable when a principal had authenticated to an authentication authority at some point in the past using any authentication context.

X509: This indicates that the principal authenticated by means of a digital signature where the key was validated as part of an X.509 Public Key Infrastructure.

PGP: This indicates that the principal authenticated by means of a digital signature where the key was validated as part of a PGP Public Key Infrastructure.

SPKI: This indicates that the principal authenticated by means of a digital signature where the key was validated via an SPKI Infrastructure.

XMLDSig: This indicates that the principal authenticated by means of a digital signature according to the processing rules specified in the XML Digital Signature specification.

Smartcard: This indicates that the principal has authenticated using smartcard.

SmartcardPKI: This class is applicable when a principal authenticates to an authentication authority through a two-factor authentication mechanism using a smartcard with enclosed private key and a PIN.

SoftwarePKI: This class is applicable when a principal uses an X.509 certificate stored in software to authenticate to the authentication authority.

Telephony: This class is used to indicate that the principal authenticated via the provision of a fixed-line telephone number, transported via a telephony protocol such as ADSL.

NomadTelephony: Indicates that the principal is "roaming" and authenticates via the means of the line number, a user suffix, and a password element.

PersonalTelephony: This class is used to indicate that the principal authenticated via the provision of a fixed-line telephone.

AuthenticatedTelephony: Indicates that the principal authenticated via the means of the line number, a user suffix, and a password element.

SecureRemotePassword: This class is applicable when the authentication was performed by means of Secure Remote Password.

TLSCClient: This class indicates that the principal authenticated by means of a client certificate, secured with the SSL/TLS transport.

TimeSyncToken: This is applicable when a principal authenticates through a time synchronization token.

Unspecified: This indicates that the authentication was performed by unspecified means.

Windows: This indicates that Windows integrated authentication is utilized for authentication.

samlBinding

This element specifies the transport mechanism of saml messages.

Possible values: REDIRECT, POST

Default value: POST

attributeConsumingServiceIndex

Index/ID of the attribute specification at Identity Provider (IdP). IdP will locate attributes requested by SP using this index and send those attributes in Assertion

Default value: 255

Minimum value: 0

Maximum value: 255

sendThumbprint

Option to send thumbprint instead of x509 certificate in SAML request

Possible values: ON, OFF

Default value: OFF

logoutURL

SingleLogout URL on IdP to which logoutRequest will be sent on Netscaler session cleanup.

rm authentication samlAction

Removes a SAML profile (action). An action cannot be removed if it is bound to a policy.

Synopsis

```
rm authentication samlAction <name>
```

Arguments

name

Name of the SAML profile to be removed.

set authentication samlAction

Modifies the specified parameters of a SAML server profile (action).

Synopsis

```
set authentication samlAction <name> [-samlIdPCertName <string>] [-samlSigningCertName <string>] [-samlRedirectUrl <string>] [-samlACSIndex <positive_integer>] [-samlUserField <string>] [-samlRejectUnsignedAssertion <samlRejectUnsignedAssertion>] [-samlIssuerName <string>] [-samlTwoFactor ( ON | OFF )] [-defaultAuthenticationGroup <string>] [-Attribute1 <string>] [-Attribute2 <string>] [-Attribute3 <string>] [-Attribute4 <string>] [-Attribute5 <string>] [-Attribute6 <string>] [-Attribute7 <string>] [-Attribute8 <string>] [-Attribute9 <string>] [-Attribute10 <string>] [-Attribute11 <string>] [-Attribute12 <string>] [-Attribute13 <string>] [-Attribute14 <string>] [-Attribute15 <string>] [-Attribute16 <string>] [-signatureAlg ( RSA-SHA1 | RSA-SHA256 )] [-digestMethod ( SHA1 | SHA256 )] [-requestedAuthnContext <requestedAuthnContext>] [-authnCtxClassRef <authnCtxClassRef> ...] [-samlBinding ( REDIRECT | POST )] [-attributeConsumingServiceIndex <positive_integer>] [-sendThumbprint ( ON | OFF )] [-logoutURL <string>]
```


Arguments

name

Name of the SAML profile (action) to modify.

samlIdPCertName

Name of the SAML server as given in that server's SSL certificate.

samlSigningCertName

Name of the signing authority as given in the SAML server's SSL certificate.

samlRedirectUrl

URL to which users are redirected for authentication.

samlACSIndex

Index/ID of the metadata entry corresponding to this configuration.

Default value: 255

Minimum value: 0

Maximum value: 255

samlUserField

SAML user ID, as given in the SAML assertion.

samlRejectUnsignedAssertion

Reject unsigned SAML assertions.

Possible values: ON, OFF, STRICT

Default value: ON

samlIssuerName

The name to be used in requests sent from Netscaler to IdP to uniquely identify Netscaler.

samlTwoFactor

Option to enable second factor after SAML

Possible values: ON, OFF

Default value: OFF

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Maximum value: 64

Attribute1

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute1

Maximum value: 64

Attribute2

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute2

Maximum value: 64

Attribute3

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute3

Maximum value: 64

Attribute4

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute4

Maximum value: 64

Attribute5

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute5

Maximum value: 64

Attribute6

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute6

Maximum value: 64

Attribute7

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute7

Maximum value: 64

Attribute8

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute8

Maximum value: 64

Attribute9

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute9

Maximum value: 64

Attribute10

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute10

Maximum value: 64

Attribute11

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute11

Maximum value: 64

Attribute12

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute12

Maximum value: 64

Attribute13

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute13

Maximum value: 64

Attribute14

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute14

Maximum value: 64

Attribute15

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute15

Maximum value: 64

Attribute16

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute16

Maximum value: 64

signatureAlg

Algorithm to be used to sign/verify SAML transactions

Possible values: RSA-SHA1, RSA-SHA256

Default value: RSA-SHA1

digestMethod

Algorithm to be used to compute/verify digest for SAML transactions

Possible values: SHA1, SHA256

Default value: SHA1

requestedAuthnContext

This element specifies the authentication context requirements of authentication statements returned in the response.

Possible values: exact, minimum, maximum, better

Default value: exact

authnCtxClassRef

This element specifies the authentication class types that are requested from IdP (IdentityProvider).

InternetProtocol: This is applicable when a principal is authenticated through the use of a provided IP address.

InternetProtocolPassword: This is applicable when a principal is authenticated through the use of a provided IP address, in addition to a username/password.

Kerberos: This is applicable when the principal has authenticated using a password to a local authentication authority, in order to acquire a Kerberos ticket.

MobileOneFactorUnregistered: This indicates authentication of the mobile device without requiring explicit end-user interaction.

MobileTwoFactorUnregistered: This indicates two-factor based authentication during mobile customer registration process, such as secure device and user PIN.

MobileOneFactorContract: Reflects mobile contract customer registration procedures and a single factor authentication.

MobileTwoFactorContract: Reflects mobile contract customer registration procedures and a two-factor based authentication.

Password: This class is applicable when a principal authenticates using password over unprotected http session.

PasswordProtectedTransport: This class is applicable when a principal authenticates to an authentication authority through the presentation of a password over a protected session.

PreviousSession: This class is applicable when a principal had authenticated to an authentication authority at some point in the past using any authentication context.

X509: This indicates that the principal authenticated by means of a digital signature where the key was validated as part of an X.509 Public Key Infrastructure.

PGP: This indicates that the principal authenticated by means of a digital signature where the key was validated as part of a PGP Public Key Infrastructure.

SPKI: This indicates that the principal authenticated by means of a digital signature where the key was validated via an SPKI Infrastructure.

XMLDSig: This indicates that the principal authenticated by means of a digital signature according to the processing rules specified in the XML Digital Signature specification.

Smartcard: This indicates that the principal has authenticated using smartcard.

SmartcardPKI: This class is applicable when a principal authenticates to an authentication authority through a two-factor authentication mechanism using a smartcard with enclosed private key and a PIN.

SoftwarePKI: This class is applicable when a principal uses an X.509 certificate stored in software to authenticate to the authentication authority.

Telephony: This class is used to indicate that the principal authenticated via the provision of a fixed-line telephone number, transported via a telephony protocol such as ADSL.

NomadTelephony: Indicates that the principal is "roaming" and authenticates via the means of the line number, a user suffix, and a password element.

PersonalTelephony: This class is used to indicate that the principal authenticated via the provision of a fixed-line telephone.

AuthenticatedTelephony: Indicates that the principal authenticated via the means of the line number, a user suffix, and a password element.

SecureRemotePassword: This class is applicable when the authentication was performed by means of Secure Remote Password.

TLSCClient: This class indicates that the principal authenticated by means of a client certificate, secured with the SSL/TLS transport.

TimeSyncToken: This is applicable when a principal authenticates through a time synchronization token.

Unspecified: This indicates that the authentication was performed by unspecified means.

Windows: This indicates that Windows integrated authentication is utilized for authentication.

samlBinding

This element specifies the transport mechanism of saml messages.

Possible values: REDIRECT, POST

Default value: POST

attributeConsumingServiceIndex

Index/ID of the attribute specification at Identity Provider (IdP). IdP will locate attributes requested by SP using this index and send those attributes in Assertion

Default value: 255

Minimum value: 0

Maximum value: 255

sendThumbprint

Option to send thumbprint instead of x509 certificate in SAML request

Possible values: ON, OFF

Default value: OFF

logoutURL

SingleLogout URL on IdP to which logoutRequest will be sent on Netscaler session cleanup.

unset authentication samlAction

Use this command to remove authentication samlAction settings. Refer to the set authentication samlAction command for meanings of the arguments.

Synopsys

```
unset authentication samlAction <name> [-samlIdPCertName] [-samlSigningCertName] [-samlRedirectUrl] [-samlACSIndex] [-samlUserField] [-samlRejectUnsignedAssertion] [-samlIssuerName] [-samlTwoFactor] [-defaultAuthenticationGroup] [-Attribute1] [-Attribute2] [-Attribute3] [-Attribute4] [-Attribute5] [-Attribute6] [-Attribute7] [-Attribute8] [-Attribute9] [-Attribute10] [-Attribute11] [-Attribute12] [-Attribute13] [-Attribute14] [-Attribute15] [-Attribute16] [-signatureAlg] [-digestMethod] [-requestedAuthnContext] [-authnCtxClassRef] [-samlBinding] [-attributeConsumingServiceIndex] [-sendThumbprint] [-logoutURL]
```

show authentication samlAction

Displays the current configuration settings for the specified SAML server profile (action).

Synopsys

```
show authentication samlAction [<name>]
```

Arguments

name

Name of the SAML server profile.

Outputs

samlIdPCertName

Name of the SAML server as given in that server's SSL certificate.

samlSigningCertName

Name of the signing authority as given in the SAML server's SSL certificate.

samlRedirectUrl

URL to which users are redirected for authentication.

samlACSIndex

Index/ID of the metadata entry corresponding to this configuration.

samlUserField

SAML user ID, as given in the SAML assertion.

samlRejectUnsignedAssertion

Reject unsigned SAML assertions.

samlIssuerName

The name to be used in requests sent from Netscaler to IdP to uniquely identify Netscaler.

samlTwoFactor

Option to enable second factor after SAML

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Attribute1

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute1

Attribute2

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute2

Attribute3

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute3

Attribute4

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute4

Attribute5

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute5

Attribute6

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute6

Attribute7

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute7

Attribute8

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute8

Attribute9

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute9

Attribute10

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute10

Attribute11

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute11

Attribute12

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute12

Attribute13

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute13

Attribute14

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute14

Attribute15

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute15

Attribute16

Name of the attribute in SAML Assertion whose value needs to be extracted and stored as attribute16

signatureAlg

Algorithm to be used to sign/verify SAML transactions

digestMethod

Algorithm to be used to compute/verify digest for SAML transactions

requestedAuthnContext

This element specifies the authentication context requirements of authentication statements returned in the response.

authnCtxClassRef

This element specifies the authentication class types that are requested from IdP (IdentityProvider).

InternetProtocol: This is applicable when a principal is authenticated through the use of a provided IP address.

InternetProtocolPassword: This is applicable when a principal is authenticated through the use of a provided IP address, in addition to a username/password.

Kerberos: This is applicable when the principal has authenticated using a password to a local authentication authority, in order to acquire a Kerberos ticket.

MobileOneFactorUnregistered: This indicates authentication of the mobile device without requiring explicit end-user interaction.

MobileTwoFactorUnregistered: This indicates two-factor based authentication during mobile customer registration process, such as secure device and user PIN.

MobileOneFactorContract: Reflects mobile contract customer registration procedures and a single factor authentication.

MobileTwoFactorContract: Reflects mobile contract customer registration procedures and a two-factor based authentication.

Password: This class is applicable when a principal authenticates using password over unprotected http session.

PasswordProtectedTransport: This class is applicable when a principal authenticates to an authentication authority through the presentation of a password over a protected session.

PreviousSession: This class is applicable when a principal had authenticated to an authentication authority at some point in the past using any authentication context.

X509: This indicates that the principal authenticated by means of a digital signature where the key was validated as part of an X.509 Public Key Infrastructure.

PGP: This indicates that the principal authenticated by means of a digital signature where the key was validated as part of a PGP Public Key Infrastructure.

SPKI: This indicates that the principal authenticated by means of a digital signature where the key was validated via an SPKI Infrastructure.

XMLDSig: This indicates that the principal authenticated by means of a digital signature according to the processing rules specified in the XML Digital Signature specification.

Smartcard: This indicates that the principal has authenticated using smartcard.

SmartcardPKI: This class is applicable when a principal authenticates to an authentication authority through a two-factor authentication mechanism using a smartcard with enclosed private key and a PIN.

SoftwarePKI: This class is applicable when a principal uses an X.509 certificate stored in software to authenticate to the authentication authority.

Telephony: This class is used to indicate that the principal authenticated via the provision of a fixed-line telephone number, transported via a telephony protocol such as ADSL.

NomadTelephony: Indicates that the principal is "roaming" and authenticates via the means of the line number, a user suffix, and a password element.

PersonalTelephony: This class is used to indicate that the principal authenticated via the provision of a fixed-line telephone.

AuthenticatedTelephony: Indicates that the principal authenticated via the means of the line number, a user suffix, and a password element.

SecureRemotePassword: This class is applicable when the authentication was performed by means of Secure Remote Password.

TLSCClient: This class indicates that the principal authenticated by means of a client certificate, secured with the SSL/TLS transport.

TimeSyncToken: This is applicable when a principal authenticates through a time synchronization token.

Unspecified: This indicates that the authentication was performed by unspecified means.

Windows: This indicates that Windows integrated authentication is utilized for authentication.

samlBinding

This element specifies the transport mechanism of saml messages.

attributeConsumingServiceIndex

Index/ID of the attribute specification at Identity Provider (IdP). IdP will locate attributes requested by SP using this index and send those attributes in Assertion

sendThumbprint

Option to send thumbprint instead of x509 certificate in SAML request

logoutURL

SingleLogout URL on IdP to which logoutRequest will be sent on Netscaler session cleanup.

devno

count

stateflag

authentication samIIdPPolicy

The following operations can be performed on "authentication samIIdPPolicy":

add | **rm** | **set** | **unset** | **show** | **stat** | **rename**

add authentication samIIdPPolicy

Adds a SAML Identity Provider (IdP) policy to use for use in authentication.

Synopsys

```
add authentication samIIdPPolicy <name> -rule <expression> -action <string> [-undefAction <string>] [-comment <string>] [-logAction <string>]
```

Arguments

name

Name for the SAML Identity Provider (IdP) authentication policy. This is used for configuring Netscaler as SAML Identity Provider. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my policy" or 'my policy').

rule

Expression which is evaluated to choose a profile for authentication.

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

The following requirements apply only to the NetScaler CLI:

- * If the expression includes one or more spaces, enclose the entire expression in double quotation marks.
- * If the expression itself includes double quotation marks, escape the quotations by using the \\ character.
- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Name of the profile to apply to requests or connections that match this policy.

undefAction

Action to perform if the result of policy evaluation is undefined (UNDEF). An UNDEF event indicates an internal error condition. Only the above built-in actions can be used.

comment

Any comments to preserve information about this policy.

logAction

Name of message log action to use when a request matches this policy.

rm authentication samIIdPPolicy

Removes an existing SAML Identity Provider (IdP) policy.

Synopsys

rm authentication samldPolicy <name>

Arguments

name

Name of the authentication policy to remove.

set authentication samldPolicy

Modifies the specified parameters of an existing SAML IdentityProvider (IdP) policy.

Synopsys

set authentication samldPolicy <name> [-rule <expression>] [-action <string>] [-undefAction <string>] [-comment <string>] [-logAction <string>]

Arguments

name

Name of the SAML Identity Provider (IdP) authentication policy to modify.

rule

Expression which is evaluated to choose a profile for authentication.

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

The following requirements apply only to the NetScaler CLI:

- * If the expression includes one or more spaces, enclose the entire expression in double quotation marks.
- * If the expression itself includes double quotation marks, escape the quotations by using the \ character.
- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Name of the profile to apply to requests or connections that match this policy.

undefAction

Action to perform if the result of policy evaluation is undefined (UNDEF). An UNDEF event indicates an internal error condition. Only the above built-in actions can be used.

comment

Any comments to preserve information about this policy.

logAction

Name of message log action to use when a request matches this policy.

unset authentication samldPolicy

Removes the settings of an existing SAML IdentityProvider (IdP) policy. Attributes for which a default value is available revert to their default values. See the set samldPolicy command for a description of the parameters..Refer to the set authentication samldPolicy command for meanings of the arguments.

Synopsys

unset authentication samlIdPPolicy <name> [-undefAction] [-comment] [-logAction]

Example

```
unset samlIdpPolicy pol9 -undefAction
```

show authentication samlIdPPolicy

Displays information about all configured SAML Identity Provider (IdP) authentication policies, or displays detailed information about the specified policy.

Synopsys

show authentication samlIdPPolicy [<name>]

Arguments

name

Name of the SAML IdentityProvider (IdP) policy for which to display detailed information.

Outputs

rule

The rule used by the SAML Identity Provider (IdP) authentication policy. Rules are combinations of Expressions. Expressions are simple conditions, such as a test for equality, applied to operands, such as a URL string or an IP address. Expression syntax is described in the Installation and Configuration Guide

action

The action to be performed when the rule is matched.

stateflag

undefAction

Action to perform if the result of policy evaluation is undefined (UNDEF). An UNDEF event indicates an internal error condition. Only the above built-in actions can be used.

comment

Any comments to preserve information about this policy.

logAction

Name of message log action to use when a request matches this policy.

boundTo

The entity name to which policy is bound

activePolicy

priority

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

hits

Number of hits.

bindPolicyType

vserverType

devno

count

stat authentication samlIdPPolicy

Display SAML Identity Provider (IdP) policy statistics.

Synopsys

```
stat authentication samlIdPPolicy [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile  
<input_filename>] [-clearstats ( basic | full )]
```

Arguments

name

The name of the SAML Identity Provider (IdP) policy for which statistics will be displayed. If not given statistics are shown for all policies.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Policy hits (Hits)

Number of hits on the policy

Example

```
stat authentication samlidppolicy.
```

rename authentication samldPPolicy

Renames the specified SAML IdentityProvider (IdP) policy. You must restart the NetScaler appliance to put new name in effect.

Synopsys

```
rename authentication samldPPolicy <name>@ <newName>@
```

Arguments

name

Existing name of the SAML IdentityProvider policy.

newName

New name for the SAML IdentityProvider policy.

Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) hash (#), space (), at (@), equals (=), colon (:), and underscore characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my samldppolicy policy" or ?my samldppolicy policy?).

Example

```
rename samldppolicy policy oldname newname
```

authentication samlIdPProfile

The following operations can be performed on "authentication samlIdPProfile":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add authentication samlIdPProfile

Creates a SAML single IdP profile. This profile is used in verifying incoming authentication request from Service Provider and creating and signing Assertion that is sent to the same.

Synopsys

```
add authentication samlIdPProfile <name> [-samlSPCertName <string>] [-samlIdPCertName <string>] [-
assertionConsumerServiceURL <URL>] [-sendPassword ( ON | OFF )] [-samlIssuerName <string>] [-
rejectUnsignedRequests ( ON | OFF )] [-signatureAlg ( RSA-SHA1 | RSA-SHA256 )] [-digestMethod ( SHA1 |
SHA256 )] [-audience <string>] [-NameIDFormat <NameIDFormat>] [-NameIDExpr <string>] [-Attribute1 <string> -
Attribute1Expr <string> [-Attribute1FriendlyName <string>] [-Attribute1Format ( URI | Basic )]] [-Attribute2 <string> -
Attribute2Expr <string> [-Attribute2FriendlyName <string>] [-Attribute2Format ( URI | Basic )]] [-Attribute3 <string> -
Attribute3Expr <string> [-Attribute3FriendlyName <string>] [-Attribute3Format ( URI | Basic )]] [-Attribute4 <string> -
Attribute4Expr <string> [-Attribute4FriendlyName <string>] [-Attribute4Format ( URI | Basic )]] [-Attribute5 <string> -
Attribute5Expr <string> [-Attribute5FriendlyName <string>] [-Attribute5Format ( URI | Basic )]] [-Attribute6 <string> -
Attribute6Expr <string> [-Attribute6FriendlyName <string>] [-Attribute6Format ( URI | Basic )]] [-Attribute7 <string> -
Attribute7Expr <string> [-Attribute7FriendlyName <string>] [-Attribute7Format ( URI | Basic )]] [-Attribute8 <string> -
Attribute8Expr <string> [-Attribute8FriendlyName <string>] [-Attribute8Format ( URI | Basic )]] [-Attribute9 <string> -
Attribute9Expr <string> [-Attribute9FriendlyName <string>] [-Attribute9Format ( URI | Basic )]] [-Attribute10 <string> -
Attribute10Expr <string> [-Attribute10FriendlyName <string>] [-Attribute10Format ( URI | Basic )]] [-Attribute11
<string> -Attribute11Expr <string> [-Attribute11FriendlyName <string>] [-Attribute11Format ( URI | Basic )]] [-
Attribute12 <string> -Attribute12Expr <string> [-Attribute12FriendlyName <string>] [-Attribute12Format ( URI | Basic
)]] [-Attribute13 <string> -Attribute13Expr <string> [-Attribute13FriendlyName <string>] [-Attribute13Format ( URI |
Basic )]] [-Attribute14 <string> -Attribute14Expr <string> [-Attribute14FriendlyName <string>] [-Attribute14Format (
URI | Basic )]] [-Attribute15 <string> -Attribute15Expr <string> [-Attribute15FriendlyName <string>] [-
Attribute15Format ( URI | Basic )]] [-Attribute16 <string> -Attribute16Expr <string> [-Attribute16FriendlyName
<string>] [-Attribute16Format ( URI | Basic )]]
```

Arguments

name

Name for the new saml single sign-on profile. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after an SSO action is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my action" or 'my action').

samlSPCertName

Name of the SSL certificate of SAML Relying Party. This certificate is used to verify signature of the incoming AuthnRequest from a Relying Party or Service Provider

samlIdPCertName

Name of the signing authority as given in the SAML server's SSL certificate. This certificate is used to sign the SAMLResponse that is sent to Relying Party or Service Provider after successful authentication

assertionConsumerServiceURL

URL to which the assertion is to be sent.

sendPassword

Option to send password in assertion.

Possible values: ON, OFF

Default value: OFF

samlIssuerName

The name to be used in requests sent from Netscaler to IdP to uniquely identify Netscaler.

rejectUnsignedRequests

Option to Reject unsigned SAML Requests.

Possible values: ON, OFF

Default value: ON

signatureAlg

Algorithm to be used to sign/verify SAML transactions

Possible values: RSA-SHA1, RSA-SHA256

Default value: RSA-SHA1

digestMethod

Algorithm to be used to compute/verify digest for SAML transactions

Possible values: SHA1, SHA256

Default value: SHA1

audience

Audience for which assertion sent by IdP is applicable. This is typically entity name or url that represents ServiceProvider

Maximum value: 256

NameIDFormat

Format of Name Identifier sent in Assertion.

Possible values: Unspecified, emailAddress, X509SubjectName, WindowsDomainQualifiedName, kerberos, entity, persistent, transient

Default value: transient

NameIDExpr

Expression that will be evaluated to obtain NameIdentifier to be sent in assertion

Attribute1

Name of attribute1 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute1Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute1FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute1Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Attribute2

Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute2Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute2FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute2Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Attribute3

Name of attribute3 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute3Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute3FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute3Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Attribute4

Name of attribute4 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute4Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute4FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute4Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Attribute5

Name of attribute5 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute5Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute5FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute5Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Attribute6

Name of attribute6 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute6Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute6FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute6Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Attribute7

Name of attribute7 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute7Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute7FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute7Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Attribute8

Name of attribute8 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute8Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute8FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute8Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Attribute9

Name of attribute9 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute9Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute9FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute9Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Attribute10

Name of attribute10 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute10Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute10FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute10Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Attribute11

Name of attribute11 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute11Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute11FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute11Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Attribute12

Name of attribute12 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute12Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute12FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute12Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Attribute13

Name of attribute13 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute13Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute13FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute13Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Attribute14

Name of attribute14 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute14Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute14FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute14Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Attribute15

Name of attribute15 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute15Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute15FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute15Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Attribute16

Name of attribute16 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute16Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute16FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute16Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

rm authentication samlIdPProfile

Deletes an existing saml IdP profile.

Synopsis

rm authentication samlIdPProfile <name>

Arguments

name

Name for the new saml single sign-on profile. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after an SSO action is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my action" or 'my action').

set authentication samlIdPProfile

Modifies the specified attributes of a saml IdP profile.

Synopsys

```
set authentication samlIdPProfile <name> [-samlSPCertName <string>] [-samlIdPCertName <string>] [-
assertionConsumerServiceURL <URL>] [-sendPassword ( ON | OFF )] [-samlIssuerName <string>] [-
rejectUnsignedRequests ( ON | OFF )] [-signatureAlg ( RSA-SHA1 | RSA-SHA256 )] [-digestMethod ( SHA1 |
SHA256 )] [-audience <string>] [-NameIDFormat <NameIDFormat>] [-NameIDExpr <string>] [-Attribute1 <string> -
Attribute1Expr <string> [-Attribute1FriendlyName <string>] [-Attribute1Format ( URI | Basic )]] [-Attribute2 <string> -
Attribute2Expr <string> [-Attribute2FriendlyName <string>] [-Attribute2Format ( URI | Basic )]] [-Attribute3 <string> -
Attribute3Expr <string> [-Attribute3FriendlyName <string>] [-Attribute3Format ( URI | Basic )]] [-Attribute4 <string> -
Attribute4Expr <string> [-Attribute4FriendlyName <string>] [-Attribute4Format ( URI | Basic )]] [-Attribute5 <string> -
Attribute5Expr <string> [-Attribute5FriendlyName <string>] [-Attribute5Format ( URI | Basic )]] [-Attribute6 <string> -
Attribute6Expr <string> [-Attribute6FriendlyName <string>] [-Attribute6Format ( URI | Basic )]] [-Attribute7 <string> -
Attribute7Expr <string> [-Attribute7FriendlyName <string>] [-Attribute7Format ( URI | Basic )]] [-Attribute8 <string> -
Attribute8Expr <string> [-Attribute8FriendlyName <string>] [-Attribute8Format ( URI | Basic )]] [-Attribute9 <string> -
Attribute9Expr <string> [-Attribute9FriendlyName <string>] [-Attribute9Format ( URI | Basic )]] [-Attribute10 <string> -
Attribute10Expr <string> [-Attribute10FriendlyName <string>] [-Attribute10Format ( URI | Basic )]] [-Attribute11
<string> -Attribute11Expr <string> [-Attribute11FriendlyName <string>] [-Attribute11Format ( URI | Basic )]] [-
Attribute12 <string> -Attribute12Expr <string> [-Attribute12FriendlyName <string>] [-Attribute12Format ( URI | Basic
)]] [-Attribute13 <string> -Attribute13Expr <string> [-Attribute13FriendlyName <string>] [-Attribute13Format ( URI |
Basic )]] [-Attribute14 <string> -Attribute14Expr <string> [-Attribute14FriendlyName <string>] [-Attribute14Format (
URI | Basic )]] [-Attribute15 <string> -Attribute15Expr <string> [-Attribute15FriendlyName <string>] [-
Attribute15Format ( URI | Basic )]] [-Attribute16 <string> -Attribute16Expr <string> [-Attribute16FriendlyName
<string>] [-Attribute16Format ( URI | Basic )]]
```

Arguments

name

Name for the new saml single sign-on profile. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after an SSO action is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my action" or 'my action').

samlSPCertName

Name of the SSL certificate of SAML Relying Party. This certificate is used to verify signature of the incoming AuthnRequest from a Relying Party or Service Provider

samlIdPCertName

Name of the signing authority as given in the SAML server's SSL certificate. This certificate is used to sign the SAMLResponse that is sent to Relying Party or Service Provider after successful authentication

assertionConsumerServiceURL

URL to which the assertion is to be sent.

sendPassword

Option to send password in assertion.

Possible values: ON, OFF

Default value: OFF

samlIssuerName

The name to be used in requests sent from Netscaler to IdP to uniquely identify Netscaler.

rejectUnsignedRequests

Option to Reject unsigned SAML Requests.

Possible values: ON, OFF

Default value: ON

signatureAlg

Algorithm to be used to sign/verify SAML transactions

Possible values: RSA-SHA1, RSA-SHA256

Default value: RSA-SHA1

digestMethod

Algorithm to be used to compute/verify digest for SAML transactions

Possible values: SHA1, SHA256

Default value: SHA1

audience

Audience for which assertion sent by IdP is applicable. This is typically entity name or url that represents ServiceProvider

Maximum value: 256

NameIDFormat

Format of Name Identifier sent in Assertion.

Possible values: Unspecified, emailAddress, X509SubjectName, WindowsDomainQualifiedName, kerberos, entity, persistent, transient

Default value: transient

NameIDExpr

Expression that will be evaluated to obtain NameIdentifier to be sent in assertion

Attribute1

Name of attribute1 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute1Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute1FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute1Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Attribute2

Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute2Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute2FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute2Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Attribute3

Name of attribute3 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute3Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute3FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute3Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Attribute4

Name of attribute4 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute4Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute4FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute4Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Attribute5

Name of attribute5 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute5Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute5FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute5Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Attribute6

Name of attribute6 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute6Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute6FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute6Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Attribute7

Name of attribute7 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute7Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute7FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute7Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Attribute8

Name of attribute8 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute8Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute8FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute8Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Attribute9

Name of attribute9 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute9Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute9FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute9Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Attribute10

Name of attribute10 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute10Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute10FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute10Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Attribute11

Name of attribute11 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute11Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute11FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute11Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Attribute12

Name of attribute12 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute12Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute12FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute12Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Attribute13

Name of attribute13 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute13Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute13FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute13Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Attribute14

Name of attribute14 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute14Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute14FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute14Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Attribute15

Name of attribute15 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute15Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute15FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute15Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Attribute16

Name of attribute16 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute16Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute16FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute16Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

unset authentication samldProfile

Use this command to remove authentication samldProfile settings. Refer to the set authentication samldProfile command for meanings of the arguments.

Synopsis

```
unset authentication samldProfile <name> [-samlSPCertName] [-samldPCertName] [-  
assertionConsumerServiceURL] [-sendPassword] [-samlIssuerName] [-rejectUnsignedRequests] [-signatureAlg] [-  
digestMethod] [-audience] [-NameIDFormat] [-NameIDExpr] [-Attribute1] [-Attribute1FriendlyName] [-  
Attribute1Format] [-Attribute2] [-Attribute2FriendlyName] [-Attribute2Format] [-Attribute3] [-Attribute3FriendlyName] [-  
Attribute3Format] [-Attribute4] [-Attribute4FriendlyName] [-Attribute4Format] [-Attribute5] [-Attribute5FriendlyName] [-  
Attribute5Format] [-Attribute6] [-Attribute6FriendlyName] [-Attribute6Format] [-Attribute7] [-Attribute7FriendlyName] [-  
Attribute7Format] [-Attribute8] [-Attribute8FriendlyName] [-Attribute8Format] [-Attribute9] [-Attribute9FriendlyName] [-  
Attribute9Format] [-Attribute10] [-Attribute10FriendlyName] [-Attribute10Format] [-Attribute11] [-  
Attribute11FriendlyName] [-Attribute11Format] [-Attribute12] [-Attribute12FriendlyName] [-Attribute12Format] [-  
Attribute13] [-Attribute13FriendlyName] [-Attribute13Format] [-Attribute14] [-Attribute14FriendlyName] [-  
Attribute14Format] [-Attribute15] [-Attribute15FriendlyName] [-Attribute15Format] [-Attribute16] [-  
Attribute16FriendlyName] [-Attribute16Format]
```

show authentication samldProfile

Displays information about all configured saml single sign-on profiles, or displays detailed information about the specified action.

Synopsis

```
show authentication samldProfile [<name>]
```

Arguments

name

Name for the new saml single sign-on profile. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after an SSO action is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my action" or 'my action').

Outputs

samlSPCertName

Name of the SSL certificate of SAML Relying Party. This certificate is used to verify signature of the incoming AuthnRequest from a Relying Party or Service Provider

samlIdPCertName

Name of the signing authority as given in the SAML server's SSL certificate. This certificate is used to sign the SAMLResponse that is sent to Relying Party or Service Provider after successful authentication

assertionConsumerServiceURL

URL to which the assertion is to be sent.

sendPassword

Option to send password in assertion.

samlIssuerName

The name to be used in requests sent from Netscaler to IdP to uniquely identify Netscaler.

rejectUnsignedRequests

Option to Reject unsigned SAML Requests.

signatureAlg

Algorithm to be used to sign/verify SAML transactions

digestMethod

Algorithm to be used to compute/verify digest for SAML transactions

audience

Audience for which assertion sent by IdP is applicable. This is typically entity name or url that represents ServiceProvider

NameIDFormat

Format of Name Identifier sent in Assertion.

NameIDExpr

Expression that will be evaluated to obtain NameIdentifier to be sent in assertion

Attribute1

Name of attribute1 that needs to be sent in SAML Assertion

Attribute2

Name of attribute2 that needs to be sent in SAML Assertion

Attribute3

Name of attribute3 that needs to be sent in SAML Assertion

Attribute4

Name of attribute4 that needs to be sent in SAML Assertion

Attribute5

Name of attribute5 that needs to be sent in SAML Assertion

Attribute6

Name of attribute6 that needs to be sent in SAML Assertion

Attribute7

Name of attribute7 that needs to be sent in SAML Assertion

Attribute8

Name of attribute8 that needs to be sent in SAML Assertion

Attribute9

Name of attribute9 that needs to be sent in SAML Assertion

Attribute10

Name of attribute10 that needs to be sent in SAML Assertion

Attribute11

Name of attribute11 that needs to be sent in SAML Assertion

Attribute12

Name of attribute12 that needs to be sent in SAML Assertion

Attribute13

Name of attribute13 that needs to be sent in SAML Assertion

Attribute14

Name of attribute14 that needs to be sent in SAML Assertion

Attribute15

Name of attribute15 that needs to be sent in SAML Assertion

Attribute16

Name of attribute16 that needs to be sent in SAML Assertion

Attribute1FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute2FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute3FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute4FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute5FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute6FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute7FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute8FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute9FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute10FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute11FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute12FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute13FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute14FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute15FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute16FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute1Format

Format of Attribute1 to be sent in Assertion.

Attribute2Format

Format of Attribute1 to be sent in Assertion.

Attribute3Format

Format of Attribute1 to be sent in Assertion.

Attribute4Format

Format of Attribute1 to be sent in Assertion.

Attribute5Format

Format of Attribute1 to be sent in Assertion.

Attribute6Format

Format of Attribute1 to be sent in Assertion.

Attribute7Format

Format of Attribute1 to be sent in Assertion.

Attribute8Format

Format of Attribute1 to be sent in Assertion.

Attribute9Format

Format of Attribute1 to be sent in Assertion.

Attribute10Format

Format of Attribute1 to be sent in Assertion.

Attribute11Format

Format of Attribute1 to be sent in Assertion.

Attribute12Format

Format of Attribute1 to be sent in Assertion.

Attribute13Format

Format of Attribute1 to be sent in Assertion.

Attribute14Format

Format of Attribute1 to be sent in Assertion.

Attribute15Format

Format of Attribute1 to be sent in Assertion.

Attribute16Format

Format of Attribute1 to be sent in Assertion.

Attribute1Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute2Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute3Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute4Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute5Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute6Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute7Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute8Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute9Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute10Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute11Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute12Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute13Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute14Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute15Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute16Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

devno

count

stateflag

authentication samlPolicy

The following operations can be performed on "authentication samlPolicy":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add authentication samlPolicy

Adds a SAML authentication policy. The policy defines the criteria under which the NetScaler appliance attempts to authenticate the user with the specified SAML server.

Synopsys

add authentication samlPolicy <name> <rule> <reqAction>

Arguments

name

Name for the SAML policy.

Must begin with a letter, number, or the underscore character (`_`), and must contain only letters, numbers, and the hyphen (`-`), period (`.`), pound (`#`), space (), at (`@`), equals (`=`), colon (`:`), and underscore characters. Cannot be changed after SAML policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, `?my authentication policy?` or `'my authentication policy?'`).

rule

Name of the NetScaler named rule, or a default syntax expression, that the policy uses to determine whether to attempt to authenticate the user with the SAML server.

reqAction

Name of the SAML authentication action to be performed if the policy matches.

rm authentication samlPolicy

Removes the specified SAML policy.

Synopsys

rm authentication samlPolicy <name>

Arguments

name

Name of the policy to remove.

set authentication samlPolicy

Modifies the specified parameters of a SAML policy.

Synopsys

set authentication samlPolicy <name> [-rule <expression>] [-reqAction <string>]

Arguments

name

Name of the SAML policy to modify.

rule

Name of the NetScaler named rule, or a default syntax expression, that the policy uses to determine whether to attempt to authenticate the user with the SAML server.

reqAction

Name of the SAML authentication action to be performed if the policy matches.

unset authentication samlPolicy

Use this command to remove authentication samlPolicy settings. Refer to the set authentication samlPolicy command for meanings of the arguments.

Synopsys

```
unset authentication samlPolicy <name> [-rule] [-reqAction]
```

show authentication samlPolicy

Displays the current settings for the specified SAML policy. If no policy name is provided, displays a list of all SAML policies currently configured on the NetScaler appliance.

Synopsys

```
show authentication samlPolicy [<name>]
```

Arguments

name

Name of the SAML policy.

Outputs

rule

The name of the new rule associated with the policy.

reqAction

The name of the SAML action associated with the policy.

boundTo

The entity name to which policy is bound

activePolicy

priority

bindPolicyType

policyType

devno

count

stateflag

authentication tacacsAction

The following operations can be performed on "authentication tacacsAction":

add | **rm** | **set** | **unset** | **show**

add authentication tacacsAction

Creates an action (profile) for a TACACS+ server. The profile contains all configuration data necessary to communicate with that TACACS+ server.

Synopsys

```
add authentication tacacsAction <name> [-serverIP <ip_addr|ipv6_addr*>] [-serverPort <port>] [-authTimeout <positive_integer>] [-tacacsSecret <secret>] [-authorization ( ON | OFF )] [-accounting ( ON | OFF )] [-auditFailedCmds ( ON | OFF )] [-defaultAuthenticationGroup <string>]
```

Arguments

name

Name for the TACACS+ profile (action).

Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.), pound (#), space (), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after TACACS profile is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my authentication action" or 'my authentication action').

serverIP

IP address assigned to the TACACS+ server.

serverPort

Port number on which the TACACS+ server listens for connections.

Default value: 49

Minimum value: 1

authTimeout

Number of seconds the NetScaler appliance waits for a response from the TACACS+ server.

Default value: 3

Minimum value: 1

tacacsSecret

Key shared between the TACACS+ server and the NetScaler appliance.

Required for allowing the NetScaler appliance to communicate with the TACACS+ server.

authorization

Use streaming authorization on the TACACS+ server.

Possible values: ON, OFF

accounting

Whether the TACACS+ server is currently accepting accounting messages.

Possible values: ON, OFF

auditFailedCmds

The state of the TACACS+ server that will receive accounting messages.

Possible values: ON, OFF

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Maximum value: 64

rm authentication tacacsAction

Removes a TACACS+ profile (action). A profile cannot be removed as long as it is bound to a policy.

Synopsis

rm authentication tacacsAction <name>

Arguments

name

Name of the profile to be removed.

set authentication tacacsAction

Modifies a TACACS+ server profile (action).

Synopsis

set authentication tacacsAction <name> [-serverIP <ip_addr|ipv6_addr|*>] [-serverPort <port>] [-authTimeout <positive_integer>] [-tacacsSecret <secret>] [-authorization (ON | OFF)] [-accounting (ON | OFF)] [-auditFailedCmds (ON | OFF)] [-defaultAuthenticationGroup <string>]

Arguments

name

Name of the TACACS+ profile to modify.

serverIP

IP address assigned to the TACACS+ server.

serverPort

Port number on which the TACACS+ server listens for connections.

Default value: 49

Minimum value: 1

authTimeout

Number of seconds the NetScaler appliance waits for a response from the TACACS+ server.

Default value: 3

Minimum value: 1

tacacsSecret

Key shared between the TACACS+ server and the NetScaler appliance.

Required for allowing the NetScaler appliance to communicate with the TACACS+ server.

authorization

Use streaming authorization on the TACACS+ server.

Possible values: ON, OFF

accounting

Whether the TACACS+ server is currently accepting accounting messages.

Possible values: ON, OFF

auditFailedCmds

The state of the TACACS+ server that will receive accounting messages.

Possible values: ON, OFF

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Maximum value: 64

unset authentication tacacsAction

Use this command to remove authentication tacacsAction settings. Refer to the set authentication tacacsAction command for meanings of the arguments.

Synopsys

```
unset authentication tacacsAction <name> [-serverIP] [-serverPort] [-authTimeout] [-tacacsSecret] [-authorization] [-accounting] [-auditFailedCmds] [-defaultAuthenticationGroup]
```

show authentication tacacsAction

Displays the current configuration settings for the specified TACACS+ profile (action).

Synopsys

```
show authentication tacacsAction [<name>]
```

Arguments

name

Name of the TACACS+ profile.

Outputs

serverIP

IP address assigned to the TACACS+ server.

serverPort

Port number on which the TACACS+ server listens for connections.

authTimeout

Number of seconds the NetScaler appliance waits for a response from the TACACS+ server.

tacacsSecret

Key shared between the TACACS+ server and the NetScaler appliance.

Required for allowing the NetScaler appliance to communicate with the TACACS+ server.

authorization

Use streaming authorization on the TACACS+ server.

accounting

Whether the TACACS+ server is currently accepting accounting messages.

auditFailedCmds

The state of the TACACS+ server that will receive accounting messages.

Success

Failure

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

stateflag

devno

count

authentication tacacsPolicy

The following operations can be performed on "authentication tacacsPolicy":

add | **rm** | **set** | **unset** | **show**

add authentication tacacsPolicy

Adds a TACACS+ authentication policy. The policy defines the criteria under which the NetScaler appliance attempts to authenticate the user with the specified TACACS+ server.

Synopsys

add authentication tacacsPolicy <name> <rule> [<reqAction>]

Arguments

name

Name for the TACACS+ policy.

Must begin with a letter, number, or the underscore character (`_`), and must contain only letters, numbers, and the hyphen (`-`), period (`.`), pound (`#`), space (), at (`@`), equals (`=`), colon (`:`), and underscore characters. Cannot be changed after TACACS+ policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, `?my authentication policy?` or `'my authentication policy?'`).

rule

Name of the NetScaler named rule, or a default syntax expression, that the policy uses to determine whether to attempt to authenticate the user with the TACACS+ server.

reqAction

Name of the TACACS+ action to perform if the policy matches.

rm authentication tacacsPolicy

Removes the specified TACACS+ policy.

Synopsys

rm authentication tacacsPolicy <name>

Arguments

name

Name of the TACACS+ policy to remove.

set authentication tacacsPolicy

Configures the specified TACACS+ policy.

Synopsys

set authentication tacacsPolicy <name> [-rule <expression>] [-reqAction <string>]

Arguments

name

Name of the TACACS+ policy.

rule

Name of the NetScaler named rule, or a default syntax expression, that the policy uses to determine whether to attempt to authenticate the user with the TACACS+ server.

reqAction

Name of the TACACS+ action to perform if the policy matches.

unset authentication tacacsPolicy

Use this command to remove authentication tacacsPolicy settings. Refer to the set authentication tacacsPolicy command for meanings of the arguments.

Synopsys

unset authentication tacacsPolicy <name> [-rule] [-reqAction]

show authentication tacacsPolicy

Displays the current settings for the specified TACACS+ policy. If no policy name is provided, displays a list of all TACACS+ policies currently configured on the NetScaler appliance.

Synopsys

show authentication tacacsPolicy [<name>]

Arguments

name

Name of the TACACS+ policy.

Outputs

rule

Name of the NetScaler named rule, or a default syntax expression, that the policy uses to determine whether to attempt to authenticate the user with the TACACS+ server.

reqAction

Name of the TACACS+ action to perform if the policy matches.

boundTo

The entity name to which policy is bound

activePolicy

priority

bindPolicyType

policyType

devno

count

stateflag

authentication vserver

The following operations can be performed on "authentication vserver":

add | **rm** | **set** | **unset** | **bind** | **unbind** | **enable** | **disable** | **show** | **stat** | **rename**

add authentication vserver

Creates an authentication virtual server.

Synopsys

```
add authentication vserver <name> <serviceType> (<IPAddress> [-range <positive_integer>]) <port> [-state (
ENABLED | DISABLED )] [-authentication ( ON | OFF )] [-AuthenticationDomain <string>] [-comment <string>] [-td
<positive_integer>] [-appflowLog ( ENABLED | DISABLED )] [-maxLoginAttempts <positive_integer>] [-
failedLoginTimeout <mins>]]
```

Arguments

name

Name for the new authentication virtual server.

Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space (), at (@), equals (=), colon (:), and underscore characters. Can be changed after the authentication virtual server is added by using the **rename authentication vserver** command.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, `?my authentication policy?` or `'my authentication policy?'`).

serviceType

Protocol type of the authentication virtual server. Always SSL.

Possible values: SSL

Default value: SSL

IPAddress

IP address of the authentication virtual server, if a single IP address is assigned to the virtual server.

range

If you are creating a series of virtual servers with a range of IP addresses assigned to them, the length of the range.

The new range of authentication virtual servers will have IP addresses consecutively numbered, starting with the primary address specified with the IP Address parameter.

Default value: 1

Minimum value: 1

port

TCP port on which the virtual server accepts connections.

Minimum value: 1

state

Initial state of the new virtual server.

Possible values: ENABLED, DISABLED

Default value: ENABLED

authentication

Require users to be authenticated before sending traffic through this virtual server.

Possible values: ON, OFF

Default value: ON

AuthenticationDomain

Fully-qualified domain name (FQDN) of the authentication virtual server.

comment

Any comments associated with this virtual server.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

appflowLog

Log AppFlow flow information.

Possible values: ENABLED, DISABLED

Default value: ENABLED

maxLoginAttempts

Maximum Number of login Attempts

Minimum value: 1

Maximum value: 255

failedLoginTimeout

Number of minutes an account will be locked if user exceeds maximum permissible attempts

Minimum value: 1

Example

The following example creates an authentication vserver named myauthenticationvip which s

rm authentication vserver

Removes an authentication virtual server.

Synopsis

rm authentication vserver <name>@ ...

Arguments

name

Name of the authentication virtual server to remove.

Example

```
rm vserver authn_vip
```

set authentication vserver

Modifies the specified parameters of an existing authentication virtual server.

Synopsys

```
set authentication vserver <name> [-IPAddress <ip_addr|ipv6_addr*>] [-authentication ( ON | OFF )] [-AuthenticationDomain <string>] [-comment <string>] [-appflowLog ( ENABLED | DISABLED )] [-maxLoginAttempts <positive_integer>] [-failedLoginTimeout <mins>]
```

Arguments

name

Name of the virtual server to modify.

IPAddress

IP address of the authentication virtual server, if a single IP address is assigned to the virtual server.

authentication

Require users to be authenticated before sending traffic through this virtual server.

Possible values: ON, OFF

Default value: ON

AuthenticationDomain

Fully-qualified domain name (FQDN) of the authentication virtual server.

comment

Any comments associated with this virtual server.

appflowLog

Log AppFlow flow information.

Possible values: ENABLED, DISABLED

Default value: ENABLED

maxLoginAttempts

Maximum Number of login Attempts

Minimum value: 1

Maximum value: 255

failedLoginTimeout

Number of minutes an account will be locked if user exceeds maximum permissible attempts

Minimum value: 1

unset authentication vserver

Removes the settings of an existing authentication virtual server. Attributes for which a default value is available revert to their default values. Refer to the set authentication vserver command for descriptions of the parameters..Refer to the set authentication vserver command for meanings of the arguments.

Synopsys

unset authentication vserver <name> [-AuthenticationDomain] [-maxLoginAttempts] [-authentication] [-comment] [-appflowLog]

bind authentication vserver

Binds authentication policies to an authentication virtual server.

Synopsys

bind authentication vserver <name> [-policy <string> [-priority <positive_integer>] [-secondary] [-groupExtraction] [-nextFactor <string>] [-gotoPriorityExpression <expression>]]

Arguments

name

Name of the authentication virtual server to which to bind the policy.

policy

Name of the policy to bind to the virtual server.

priority

Positive integer specifying the priority of the policy. A lower number specifies a higher priority. Policies are evaluated in the order of their priorities, and the first policy that matches the request is applied. Must be unique within the list of policies bound to the authentication virtual server.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my authentication policy? or ?my authentication policy?).

Minimum value: 0

secondary

Applicable only while binding classic authentication policy as advance authentication policy use nFactor

groupExtraction

Applicable only while binding classic authentication policy as advance authentication policy use nFactor

nextFactor

Applicable only while binding advance authentication policy as classic authentication policy does not support nFactor

gotoPriorityExpression

Applicable only to advance authentication policy. Expression or other value specifying the next policy to be evaluated if the current policy evaluates to TRUE. Specify one of the following values:

- * NEXT - Evaluate the policy with the next higher priority number.
- * END - End policy evaluation.
- * USE_INVOCATION_RESULT - Applicable if this policy invokes another policy label. If the final goto in the invoked policy label has a value of END, the evaluation stops. If the final goto is anything other than END, the current policy label performs a NEXT.
- * A default syntax expression that evaluates to a number.

If you specify an expression, the number to which it evaluates determines the next policy to evaluate, as follows:

- * If the expression evaluates to a higher numbered priority, the policy with that priority is evaluated next.

* If the expression evaluates to the priority of the current policy, the policy with the next higher numbered priority is evaluated next.

* If the expression evaluates to a priority number that is numerically higher than the highest numbered priority, policy evaluation ends.

An UNDEF event is triggered if:

* The expression is invalid.

* The expression evaluates to a priority number that is numerically lower than the current policy's priority.

* The expression evaluates to a priority number that is between the current policy's priority number (say, 30) and the highest priority number (say, 100), but does not match any configured priority number (for example, the expression evaluates to the number 85). This example assumes that the priority number increments by 10 for every successive policy, and therefore a priority number of 85 does not exist in the policy label.

unbind authentication vserver

Unbinds the specified policy from the specified authentication virtual server.

Synopsys

```
unbind authentication vserver <name> [-policy <string> [-secondary] [-groupExtraction]]
```

Arguments

name

Name of the virtual server.

policy

Name of the policy to be unbound.

secondary

Applicable only to classic authentication policy

groupExtraction

Applicable only to classic authentication policy

enable authentication vserver

Enables an authentication virtual server that is disabled. Note: Virtual servers, when added, are normally enabled by default.

Synopsys

```
enable authentication vserver <name>@
```

Arguments

name

Name of the virtual server to enable.

Example

```
enable vserver authentication1
```

disable authentication vserver

Disables an authentication virtual server, taking it out of service.

Synopsys

disable authentication vserver <name>@

Arguments

name

Name of the virtual server to disable.

Notes:

1. The NetScaler appliance still responds to ARP and/or ping requests for the IP address of disabled virtual servers.
2. Because the virtual server configuration still exists on the NetScaler appliance, you can reenable the virtual server.

Example

```
disable vserver authn_vip
```

show authentication vserver

Displays the configuration of the specified authentication virtual server. If no authentication virtual server is specified, displays a list of all authentication virtual servers that are currently configured on the NetScaler appliance.

Synopsys

show authentication vserver [<name>] show authentication vserver stats - alias for 'stat authentication vserver'

Arguments

name

Name of the authentication virtual server.

Outputs

IPAddress

The IP address of the authentication server.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

value

Indicates whether or not the certificate is bound or if SSL offload is disabled.

port

The virtual TCP port of the authentication vserver.

range

The range of authentication vserver IP addresses. The new range of authentication vservers will have IP addresses consecutively numbered, starting with the primary address specified with the <ipaddress> argument.

serviceType

The authentication vserver's protocol type, Currently the only possible value is SSL.

type

The type of Virtual Server, e.g. CONTENT based or ADDRESS based.

state

Initial state of the new virtual server.

status

Whether or not this vserver responds to ARPs and whether or not round-robin selection is temporarily in effect.

cacheType

Virtual server's cache type. The options are: TRANSPARENT, REVERSE and FORWARD.

redirect

The cache redirect policy.

The valid redirect policies are:

1. CACHE - Directs all requests to the cache.
2. POLICY - Applies cache redirection policy to determine whether the request should be directed to the cache or origin. This is the default setting.
3. ORIGIN - Directs all requests to the origin server.

precedence

This argument is used only when configuring content switching on the specified virtual server. This is applicable only

if both the URL and RULE-based policies have been configured on the same virtual server.

It specifies the type of policy (URL or RULE) that takes precedence on the content switching virtual server. The default setting is RULE.

I URL - In this case, the incoming request is matched against the URL-based policies before the rule-based policies.

I RULE - In this case, the incoming request is matched against the rule-based policies before the URL-based policies.

For all URL-based policies, the precedence hierarchy is:

1. Domain and exact URL
2. Domain, prefix and suffix
3. Domain and suffix
4. Domain and prefix
5. Domain only
6. Exact URL
7. Prefix and suffix
8. Suffix only
9. Prefix only
10. Default

redirectURL

The URL where traffic is redirected if the virtual server in system becomes unavailable. WARNING! Make sure that the domain you specify in the URL does not match the domain specified in the -d domainName argument of the ###add cs policy### command. If the same domain is specified in both arguments, the request will be continuously redirected to the same unavailable virtual server in the system. If so, the user may not get the requested content.

authentication

Indicates whether or not authentication is being applied to incoming users to the VPN.

curAAAUsers

The number of current users logged in to this vserver.

AuthenticationDomain

Fully-qualified domain name (FQDN) of the authentication virtual server.

rule

The name of the rule, or expression, if any, that policy for the authentication server is to use. Rules are combinations of Expressions. Expressions are simple conditions, such as a test for equality, applied to operands, such as a URL string or an IP address. Expression syntax is described in the Installation and Configuration Guide. The default rule is ns_true.

policyName

The name of the policy, if any, bound to the authentication vserver.

policy

The name of the policy, if any, bound to the authentication vserver.

serviceName

The name of the service, if any, to which the vserver policy is bound.

weight

Weight for this service, if any. This weight is used when the system performs load balancing, giving greater priority to a specific service. It is useful when the services bound to a virtual server are of different capacity.

cacheVserver

The name of the default target cache virtual server, if any, to which requests are redirected.

backupVServer

The name of the backup vpn virtual server for this vpn virtual server.

cltTimeout

The idle time, if any, in seconds after which the client connection is terminated.

soMethod

VPN client applications are allocated from a block of Intranet IP addresses.

That block may be exhausted after a certain number of connections. This switch specifies the method used to determine whether or not a new connection will spillover, or exhaust, the allocated block of Intranet IP addresses for that application. Possible values are CONNECTION or DYNAMICCONNECTION. CONNECTION means that a static integer value is the hard limit for the spillover threshold. The spillover threshold is described below. DYNAMICCONNECTION means that the spillover threshold is set according to the maximum number of connections defined for the vpn vserver.

soThreshold

VPN client applications are allocated from a block of Intranet IP addresses.

That block may be exhausted after a certain number of connections.

The value of this option is number of client connections after which the Mapped IP address is used as the client source IP address instead of an address from the allocated block of Intranet IP addresses.

soPersistence

Whether or not cookie-based site persistence is enabled for this VPN vserver. Possible values are 'ConnectionProxy', HTTPRedirect, or NONE

soPersistenceTimeout

The timeout, if any, for cookie-based site persistence of this VPN vserver.

priority

The priority, if any, of the vpn vserver policy.

downStateFlush

Perform delayed clean up of connections on this vserver.

actType

disablePrimaryOnDown

Tells whether traffic will continue reaching backup vservers even after primary comes UP from DOWN state.

Listenpolicy

Listenpolicy configured for authentication vserver

Listenpriority

Priority of listen policy for authentication vserver

tcpProfileName

The name of the TCP profile.

httpProfileName

Name of the HTTP profile.

comment

Any comments associated with this virtual server.

policySubType

stateflag

flags

appflowLog

Log AppFlow flow information.

vstype

Virtual Server Type, e.g. Load Balancing, Content Switch, Cache Redirection

ngname

Nodegroup devno to which this authentication vserver belongs to

maxLoginAttempts

Maximum Number of login Attempts

failedLoginTimeout

Number of minutes an account will be locked if user exceeds maximum permissible attempts

secondary

Bind the authentication policy to the secondary chain.

Provides for multifactor authentication in which a user must authenticate via both a primary authentication method and, afterward, via a secondary authentication method.

Because user groups are aggregated across authentication systems, usernames must be the same on all authentication servers. Passwords can be different.

groupExtraction

Bind the Authentication policy to a tertiary chain which will be used only for group extraction. The user will not authenticate against this server, and this will only be called if primary and/or secondary authentication has succeeded.

nextFactor

On success invoke label.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

devno

count

Example

```
show authentication vserver
```

stat authentication vserver

Displays statistics about the specified authentication virtual server. If no authentication virtual server is specified, displays statistics for all authentication virtual servers that are currently configured on the NetScaler appliance.

Synopsys

```
stat authentication vserver [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

name

Name of the authentication virtual server.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

IP address (IP)

The IP address on which the service is running.

Port (port)

The port on which the service is running.

Vserver protocol (Protocol)

Protocol associated with the vservers

State

Current state of the server. Possible values are UP, DOWN, UNKNOWN, OFS(Out of Service), TROFS (Transition Out of Service), TROFS_DOWN(Down When going Out of Service)

Requests (Req)

Total number of requests received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

Responses (Rsp)

Number of responses received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

Request bytes (Reqb)

Total number of request bytes received on this service or virtual server.

Response bytes (Rspb)

Number of response bytes received by this service or virtual server.

rename authentication vservers

Rename an authentication virtual server.

Synopsis

rename authentication vservers <name>@ <newName>@

Arguments

name

Current name of the authentication virtual server.

newName

New name of the authentication virtual server.

Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space (), at (@), equals (=), colon (:), and underscore characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my authentication policy" or 'my authentication policy').

Example

```
rename authentication vserver av1 av_new
```

authentication webAuthAction

The following operations can be performed on "authentication webAuthAction":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add authentication webAuthAction

Adds an action to be used for web authentication. * Specify the entire HTTP request in a single expression.

Synopsys

```
add authentication webAuthAction <name> -serverIP <ip_addr|ipv6_addr|*> -serverPort <port|*> [-fullReqExpr <string>] -scheme ( http | https ) -successRule <expression> [-defaultAuthenticationGroup <string>] [-Attribute1 <string>] [-Attribute2 <string>] [-Attribute3 <string>] [-Attribute4 <string>] [-Attribute5 <string>] [-Attribute6 <string>] [-Attribute7 <string>] [-Attribute8 <string>] [-Attribute9 <string>] [-Attribute10 <string>] [-Attribute11 <string>] [-Attribute12 <string>] [-Attribute13 <string>] [-Attribute14 <string>] [-Attribute15 <string>] [-Attribute16 <string>]
```

Arguments

name

Name for the Web Authentication action.

Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space (), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after the profile is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my authentication action? or ?my authentication action?).

serverIP

IP address of the web server to be used for authentication.

serverPort

Port on which the web server accepts connections.

Minimum value: 1

fullReqExpr

Exact HTTP request, in the form of a default syntax expression, which the NetScaler appliance sends to the authentication server.

The NetScaler appliance does not check the validity of this request. One must manually validate the request.

scheme

Type of scheme for the web server.

Possible values: http, https

successRule

Expression, that checks to see if authentication is successful.

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Maximum value: 64

Attribute1

Expression that would be evaluated to extract attribute1 from the webauth response

Attribute2

Expression that would be evaluated to extract attribute2 from the webauth response

Attribute3

Expression that would be evaluated to extract attribute3 from the webauth response

Attribute4

Expression that would be evaluated to extract attribute4 from the webauth response

Attribute5

Expression that would be evaluated to extract attribute5 from the webauth response

Attribute6

Expression that would be evaluated to extract attribute6 from the webauth response

Attribute7

Expression that would be evaluated to extract attribute7 from the webauth response

Attribute8

Expression that would be evaluated to extract attribute8 from the webauth response

Attribute9

Expression that would be evaluated to extract attribute9 from the webauth response

Attribute10

Expression that would be evaluated to extract attribute10 from the webauth response

Attribute11

Expression that would be evaluated to extract attribute11 from the webauth response

Attribute12

Expression that would be evaluated to extract attribute12 from the webauth response

Attribute13

Expression that would be evaluated to extract attribute13 from the webauth response

Attribute14

Expression that would be evaluated to extract attribute14 from the webauth response

Attribute15

Expression that would be evaluated to extract attribute15 from the webauth response

Attribute16

Expression that would be evaluated to extract attribute16 from the webauth response

Example

```
add authentication webAuthAction al -ServerIP 1.1.1.1 -ServerPort 80 -scheme HTTP -success:
```

rm authentication webAuthAction

Removes a web authentication action. You cannot remove an action that is used in any part of a policy.

Synopsys

rm authentication webAuthAction <name>

Arguments

name

Name of the web authentication action to remove.

Example

```
rm authentication webAuthAction al
```

set authentication webAuthAction

Modifies the attributes of an existing web authentication action.

Synopsys

```
set authentication webAuthAction <name> [-serverIP <ip_addr|ipv6_addr|*>] [-serverPort <port|*>] [-fullReqExpr <string>] [-scheme ( http | https )] [-successRule <expression>] [-defaultAuthenticationGroup <string>] [-Attribute1 <string>] [-Attribute2 <string>] [-Attribute3 <string>] [-Attribute4 <string>] [-Attribute5 <string>] [-Attribute6 <string>] [-Attribute7 <string>] [-Attribute8 <string>] [-Attribute9 <string>] [-Attribute10 <string>] [-Attribute11 <string>] [-Attribute12 <string>] [-Attribute13 <string>] [-Attribute14 <string>] [-Attribute15 <string>] [-Attribute16 <string>]
```

Arguments

name

Name of the action to configure.

serverIP

IP address of the web server to be used for authentication.

serverPort

Port on which the web server accepts connections.

Minimum value: 1

fullReqExpr

Exact HTTP request, in the form of a default syntax expression, which the NetScaler appliance sends to the authentication server.

The NetScaler appliance does not check the validity of this request. One must manually validate the request.

scheme

Type of scheme for the web server.

Possible values: http, https

successRule

Expression, that checks to see if authentication is successful.

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Maximum value: 64

Attribute1

Expression that would be evaluated to extract attribute1 from the webauth response

Attribute2

Expression that would be evaluated to extract attribute2 from the webauth response

Attribute3

Expression that would be evaluated to extract attribute3 from the webauth response

Attribute4

Expression that would be evaluated to extract attribute4 from the webauth response

Attribute5

Expression that would be evaluated to extract attribute5 from the webauth response

Attribute6

Expression that would be evaluated to extract attribute6 from the webauth response

Attribute7

Expression that would be evaluated to extract attribute7 from the webauth response

Attribute8

Expression that would be evaluated to extract attribute8 from the webauth response

Attribute9

Expression that would be evaluated to extract attribute9 from the webauth response

Attribute10

Expression that would be evaluated to extract attribute10 from the webauth response

Attribute11

Expression that would be evaluated to extract attribute11 from the webauth response

Attribute12

Expression that would be evaluated to extract attribute12 from the webauth response

Attribute13

Expression that would be evaluated to extract attribute13 from the webauth response

Attribute14

Expression that would be evaluated to extract attribute14 from the webauth response

Attribute15

Expression that would be evaluated to extract attribute15 from the webauth response

Attribute16

Expression that would be evaluated to extract attribute16 from the webauth response

Example

```
set authentication webAuthAction al -ServerIP 1.1.1.1 -ServerPort 80
```

unset authentication webAuthAction

Use this command to remove authentication webAuthAction settings. Refer to the set authentication webAuthAction command for meanings of the arguments.

Synopsys

```
unset authentication webAuthAction <name> [-serverIP] [-serverPort] [-fullReqExpr] [-defaultAuthenticationGroup] [-Attribute1] [-Attribute2] [-Attribute3] [-Attribute4] [-Attribute5] [-Attribute6] [-Attribute7] [-Attribute8] [-Attribute9] [-Attribute10] [-Attribute11] [-Attribute12] [-Attribute13] [-Attribute14] [-Attribute15] [-Attribute16]
```

show authentication webAuthAction

Displays information about the configured web authentication action.

Synopsys

```
show authentication webAuthAction [<name>]
```

Arguments

name

Name of the web authentication action to display. If a name is not provided, information about all actions is shown.

Outputs

stateflag

serverIP

IP address of the web server to be used for authentication.

serverPort

Port on which the web server accepts connections.

fullReqExpr

Exact HTTP request, in the form of a default syntax expression, which the NetScaler appliance sends to the authentication server.

The NetScaler appliance does not check the validity of this request. One must manually validate the request.

scheme

Type of scheme for the web server.

successRule

Expression, that checks to see if authentication is successful.

defaultAuthenticationGroup

This is the default group that is chosen when the authentication succeeds in addition to extracted groups.

Attribute1

Expression that would be evaluated to extract attribute1 from the webauth response

Attribute2

Expression that would be evaluated to extract attribute2 from the webauth response

Attribute3

Expression that would be evaluated to extract attribute3 from the webauth response

Attribute4

Expression that would be evaluated to extract attribute4 from the webauth response

Attribute5

Expression that would be evaluated to extract attribute5 from the webauth response

Attribute6

Expression that would be evaluated to extract attribute6 from the webauth response

Attribute7

Expression that would be evaluated to extract attribute7 from the webauth response

Attribute8

Expression that would be evaluated to extract attribute8 from the webauth response

Attribute9

Expression that would be evaluated to extract attribute9 from the webauth response

Attribute10

Expression that would be evaluated to extract attribute10 from the webauth response

Attribute11

Expression that would be evaluated to extract attribute11 from the webauth response

Attribute12

Expression that would be evaluated to extract attribute12 from the webauth response

Attribute13

Expression that would be evaluated to extract attribute13 from the webauth response

Attribute14

Expression that would be evaluated to extract attribute14 from the webauth response

Attribute15

Expression that would be evaluated to extract attribute15 from the webauth response

Attribute16

Expression that would be evaluated to extract attribute16 from the webauth response

devno

count

Example

```
show authentication webAuthAction al
```

authentication webAuthPolicy

The following operations can be performed on "authentication webAuthPolicy":

[add](#) | [rm](#) | [set](#) | [show](#)

add authentication webAuthPolicy

Adds an WebAuth authentication policy. The policy defines the criteria under which the NetScaler appliance attempts to authenticate the user with the specified Web server.

Synopsys

```
add authentication webAuthPolicy <name> -rule <string> -action <string>
```

Arguments

name

Name for the WebAuth policy.

Must begin with a letter, number, or the underscore character (`_`), and must contain only letters, numbers, and the hyphen (`-`), period (`.`) pound (`#`), space (), at (`@`), equals (`=`), colon (`:`), and underscore characters. Cannot be changed after LDAP policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, `?my authentication policy?` or `'my authentication policy?'`).

rule

Name of the NetScaler named rule, or a default syntax expression, that the policy uses to determine whether to attempt to authenticate the user with the Web server.

action

Name of the WebAuth action to perform if the policy matches.

rm authentication webAuthPolicy

Removes an WebAuth policy.

Synopsys

```
rm authentication webAuthPolicy <name>
```

Arguments

name

Name of the WebAuth policy to remove.

set authentication webAuthPolicy

Configures the specified WebAuth policy.

Synopsys

```
set authentication webAuthPolicy <name> [-rule <string>] [-action <string>]
```

Arguments

name

Name of the WebAuth policy.

rule

The new rule to associate with the policy.

action

The new WebAuth action to associate with the policy.

show authentication webAuthPolicy

Displays the current settings for the specified WebAuth policy. If no policy name is provided, displays a list of all WebAuth policies currently configured on the NetScaler appliance.

Synopsys

show authentication webAuthPolicy [<name>]

Arguments

name

Name of the WebAuth policy.

Outputs

rule

Name of the NetScaler named rule, or a default syntax expression, that the policy uses to determine whether to attempt to authenticate the user with the Web server.

action

Name of the WebAuth action to perform if the policy matches.

boundTo

The entity name to which policy is bound

activePolicy

priority

bindPolicyType

policyType

devno

count

stateflag

Authorization Commands

The entities on which you can perform NetScaler CLI operations:

- `authorization action`
- `authorization policy`
- `authorization policylabel`

authorization action

The following operations can be performed on "authorization action":

show authorization action

Show details of authorization actions.

Synopsys

show authorization action [<name>]

Arguments

name

Name of authorization action

Outputs

devno

count

stateflag

authorization policy

The following operations can be performed on "authorization policy":

[add](#) | [rm](#) | [set](#) | [rename](#) | [show](#)

add authorization policy

Creates an authorization policy. Authorization policies allow AAA users and AAA groups to access resources through SSL VPN/AAA-TM enabled virtual servers.

Synopsys

add authorization policy <name> <rule> <action>

Arguments

name

Name for the new authorization policy.

Must begin with a letter, number, or the underscore character (`_`), and must contain only letters, numbers, and the hyphen (`-`), period (`.`), pound (`#`), space (), at (`@`), equals (`=`), colon (`:`), and underscore characters. Cannot be changed after the authorization policy is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, `?my authorization policy?` or `'my authorization policy?'`).

rule

Name of the NetScaler named rule, or a default syntax expression, that the policy uses to perform the authentication.

action

Action to perform if the policy matches: either allow or deny the request.

Example

Example: Consider the following authorization policy, "author-policy", `add authorization`

rm authorization policy

Removes an authorization policy.

Synopsys

rm authorization policy <name>

Arguments

name

Name of the authorization policy to be removed.

set authorization policy

Configures the specified parameters of an authorization policy.

Synopsys

set authorization policy <name> [-rule <expression>] [-action <string>]

Arguments

name

Name of the authorization policy to modify.

rule

Name of the NetScaler named rule, or a default syntax expression, that the policy uses to perform the authentication.

action

Action to perform if the policy matches: either allow or deny the request.

rename authorization policy

Rename a author policy.

Synopsys

rename authorization policy <name>@ <newName>@

Arguments

name

The name of the author policy.

newName

The new name of the author policy.

Example

```
rename auth policy oldname newname
```

show authorization policy

Displays the current settings for the specified authorization policy. If no policy name is provided, displays a list of all authorization policies currently configured on the NetScaler appliance.

Synopsys

show authorization policy [<name>]

Arguments

name

Name of the authorization policy.

Outputs

rule

Rule of the policy.

action

Authorization action associated with the policy. It can be either ALLOW or DENY.

boundTo

The entity name to which policy is bound

activePolicy

priority

flag

bindPolicyType

policyType

vserverType

expressionType

Type of policy (Classic/Advanced)

devno

count

stateflag

authorization policylabel

The following operations can be performed on "authorization policylabel":

add | **rm** | **bind** | **unbind** | **rename** | **show** | **stat**

add authorization policylabel

Creates a user-defined authorization policy label.

Synopsys

add authorization policylabel <labelName>

Arguments

labelName

Name for the new authorization policy label.

Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space (), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after the authorization policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my authorization policy label? or ?authorization policy label?).

Example

```
add authorization policylabel trans_http_url
```

rm authorization policylabel

Removes an authorization policy label.

Synopsys

rm authorization policylabel <labelName>

Arguments

labelName

Name of the authorization policy label to remove.

Example

```
rm authorization policylabel trans_http_url
```

bind authorization policylabel

Binds an authorization policy to a label.

Synopsys

bind authorization policylabel <labelName> <policyName> <priority> [<gotoPriorityExpression>] [-invoke
(<labelType> <labelName>)]

Arguments

labelName

Name of the policy label to invoke if the current policy evaluates to TRUE, the invoke parameter is set, and Label Type is set to Policy Label.

policyName

Name of the authorization policy to bind to the policy label.

priority

Positive integer specifying the priority of the policy. The lower the number, the higher the priority. Policies within a label are evaluated in the order of their priority numbers.

Minimum value: 1

Maximum value: 2147483647

gotoPriorityExpression

Expression or other value specifying the next policy to be evaluated if the current policy evaluates to TRUE. Specify one of the following values:

- * NEXT ? Evaluate the policy with the next higher priority number.
- * END ? End policy evaluation.
- * USE_INVOCATION_RESULT ? Applicable if this policy invokes another policy label. If the final goto in the invoked policy label has a value of END, the evaluation stops. If the final goto is anything other than END, the current policy label performs a NEXT.
- * A default syntax or classic expression that evaluates to a number.

If you specify an expression, the number to which it evaluates determines the next policy to evaluate, as follows:

- * If the expression evaluates to a higher numbered priority, the policy with that priority is evaluated next.
- * If the expression evaluates to the priority of the current policy, the policy with the next higher numbered priority is evaluated next.
- * If the expression evaluates to a number that is larger than the largest numbered priority, policy evaluation ends.

An UNDEF event is triggered if:

- * The expression is invalid.
- * The expression evaluates to a priority number that is smaller than the current policy's priority number.
- * The expression evaluates to a priority number that is between the current policy's priority number (say, 30) and the highest priority number (say, 100), but does not match any configured priority number (for example, the expression evaluates to the number 85). This example assumes that the priority number increments by 10 for every successive policy, and therefore a priority number of 85 does not exist in the policy label.

invoke

If the current policy evaluates to TRUE, terminate evaluation of policies bound to the current policy label, and then either forward the request or response to the specified virtual server or evaluate the specified policy label.

labelType

Type of invocation. Available settings function as follows:

- * reqvserver - Send the request to the specified request virtual server.
- * resvserver - Send the response to the specified response virtual server.
- * policylabel - Invoke the specified policy label.

Possible values: reqvserver, resvserver, policylabel

Example

```
i) bind authorization policylabel trans_http_url pol_1 1 2 -invoke reqvserver CURRENT i:
```

unbind authorization policylabel

Unbinds the specified policy from the specified authorization policy label.

Synopsys

```
unbind authorization policylabel <labelName> <policyName> [-priority <positive_integer>]
```

Arguments

labelName

Name for the new authorization policy label.

Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space (), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after the authorization policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my authorization policy label? or ?authorization policy label?).

policyName

Name of the authorization policy to bind to the policy label.

priority

Priority of the NOPOLICY to be unbound.

Minimum value: 1

Maximum value: 2147483647

Example

```
unbind authorization policylabel trans_http_url pol_1
```

rename authorization policylabel

Rename a auth policy label.

Synopsys

```
rename authorization policylabel <labelName>@ <newName>@
```

Arguments

labelName

The name of the auth policy label

newName

The new name of the auth policy label

Example

```
rename auth policy label oldname newname
```

show authorization policylabel

Displays the current settings for the specified authorization policy label. If no policy name is provided, displays a list of all authorization policy labels currently configured on the NetScaler appliance.

Synopsys

```
show authorization policylabel [<labelName>]
```

Arguments

labelName

Name of the authorization policy label.

Outputs

stateflag

numpol

Number of policies bound to label.

hits

Number of times policy label was invoked.

policyName

Name of the authorization policy to bind to the policy label.

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

labelType

Type of invocation. Available settings function as follows:

- * reqvserver - Send the request to the specified request virtual server.
- * resvserver - Send the response to the specified response virtual server.
- * policylabel - Invoke the specified policy label.

labelName

Name of the policy label to invoke if the current policy evaluates to TRUE, the invoke parameter is set, and Label Type is set to Policy Label.

flowType

Flowtype of the bound authorization policy.

description

Description of the policylabel

flags

devno

count

Example

```
i) show authorization policylabel trans_http_url ii) show authorization policylabel
```

stat authorization policylabel

Displays statistics for the specified authorization policy label. If no authorization policy label is specified, displays a list of all authorization policy labels.

Synopsys

```
stat authorization policylabel [<labelName>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile  
<input_filename>] [-clearstats ( basic | full )]
```

Arguments

labelName

Name of the authorization policy label.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Policy Label Hits (Hits)

Number of times policy label was invoked.

AutoScale Commands

The entities on which you can perform NetScaler CLI operations:

- [autoscale action](#)
- [autoscale policy](#)
- [autoscale profile](#)

autoscale action

The following operations can be performed on "autoscale action":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add autoscale action

Create a AutoScale action.

Synopsys

```
add autoscale action <name> -type ( SCALE_UP | SCALE_DOWN ) -profileName <string> -parameters <string> [-vmDestroyGracePeriod <positive_integer>] [-quietTime <positive_integer>] -vServer <string>
```

Arguments

name

ActionScale action name.

type

The type of action.

Possible values: SCALE_UP, SCALE_DOWN

profileName

AutoScale profile name.

parameters

Parameters to use in the action

vmDestroyGracePeriod

Time in minutes a VM is kept in inactive state before destroying

Default value: 10

Minimum value: 0

quietTime

Time in seconds no other policy is evaluated or action is taken

Default value: 300

Minimum value: 0

vServer

Name of the vserver on which autoscale action has to be taken.

rm autoscale action

Remove a AutoScale action.

Synopsys

```
rm autoscale action <name>
```

Arguments

name

ActionScale action name.

set autoscale action

Set a AutoScale action.

Synopsys

```
set autoscale action <name> [-profileName <string>] [-parameters <string>] [-vmDestroyGracePeriod  
<positive_integer>] [-quietTime <positive_integer>] [-vServer <string>]
```

Arguments

name

ActionScale action name.

profileName

AutoScale profile name.

parameters

Parameters to use in the action

vmDestroyGracePeriod

Time in minutes a VM is kept in inactive state before destroying

Default value: 10

Minimum value: 0

quietTime

Time in seconds no other policy is evaluated or action is taken

Default value: 300

Minimum value: 0

vServer

Name of the vserver on which autoscale action has to be taken.

unset autoscale action

Use this command to remove autoscale action settings.Refer to the set autoscale action command for meanings of the arguments.

Synopsys

```
unset autoscale action <name> [-vmDestroyGracePeriod] [-quietTime]
```

show autoscale action

Display the autoscale actions.

Synopsys

```
show autoscale action [<name>]
```

Arguments

name

ActionScale action name.

Outputs

type

The type of action.

profileName

AutoScale profile name.

parameters

Parameters to use in the action

vmDestroyGracePeriod

Time in minutes a VM is kept in inactive state before destroying

quietTime

Time in seconds no other policy is evaluated or action is taken

vServer

Name of the vserver on which autoscale action has to be taken.

destIP

IP Address on which provisioning server daemon is running

destPort

Port on which provisioning server daemon is running

devno

count

stateflag

autoscale policy

The following operations can be performed on "autoscale policy":

add | **rm** | **set** | **unset** | **show** | **stat** | **rename**

add autoscale policy

Create a autoscale policy.

Synopsis

```
add autoscale policy <name> -rule <expression> -action <string> [-comment <string>] [-logAction <string>]
```

Arguments

name

The name of the autoscale policy.

rule

The rule associated with the policy.

action

The autoscale profile associated with the policy.

comment

Comments associated with this autoscale policy.

logAction

The log action associated with the autoscale policy

rm autoscale policy

Remove a autoscale policy.

Synopsis

```
rm autoscale policy <name>
```

Arguments

name

The name of the autoscale policy.

Example

```
rm autoscale policy pol
```

set autoscale policy

Set a new rule/action/comment for an existing autoscale policy.

Synopsis

```
set autoscale policy <name> [-rule <expression>] [-action <string>] [-comment <string>] [-logAction <string>]
```

Arguments

name

The name of the autoscale policy.

rule

The rule associated with the policy.

action

The autoscale profile associated with the policy.

comment

Comments associated with this autoscale policy.

logAction

The log action associated with the autoscale policy

Example

```
set autoscaler policy pol -rule true
```

unset autoscale policy

Unset comment/logaction for existing autoscale policy..Refer to the set autoscale policy command for meanings of the arguments.

Synopsys

```
unset autoscale policy <name> [-rule <expression>] [-action <string>] [-comment <string>] [-logAction <string>]
```

Example

```
unset autoscale policy pol9 -undefAction
```

show autoscale policy

Display the autoscale policies.

Synopsys

```
show autoscale policy [<name>]
```

Arguments

name

The name of the autoscale policy.

Outputs

rule

The rule associated with the policy.

action

The autoscale profile associated with the policy.

comment

Comments associated with this autoscale policy.

logAction

The log action associated with the autoscale policy

stateflag**hits**

Number of hits.

undefHits

Number of Undef hits.

priority

Specifies the priority of the policy.

boundTo

Location where policy is bound

activePolicy

Indicates whether policy is bound or not.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

devno**count**

stat autoscale policy

Display autoscale policy statistics.

Synopsys

```
stat autoscale policy [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

name

The name of the autoscale policy for which statistics will be displayed. If not given statistics are shown for all autoscale policies.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Policy hits (Hits)

Number of hits on the policy

Policy undef hits (Undefhits)

Number of undef hits on the policy

Example

```
stat autoscale policy
```

rename autoscale policy

Rename a autoscale policy.

Synopsys

```
rename autoscale policy <name>@ <newName>@
```

Arguments

name

The name of the autoscale policy.

newName

The new name of the autoscale policy.

Example

```
rename autoscale policy oldname newname
```

autoscale profile

The following operations can be performed on "autoscale profile":

[add](#) | [rm](#) | [set](#) | [show](#)

add autoscale profile

Create a AutoScale policy.

Synopsys

```
add autoscale profile <name> -type CLOUDSTACK -url <URL> -apiKey -sharedSecret
```

Arguments

name

AutoScale profile name.

type

The type of profile.

Possible values: CLOUDSTACK

url

URL providing the service

apiKey

api key for authentication with service

sharedSecret

shared secret for authentication with service

rm autoscale profile

Remove a AutoScale policy.

Synopsys

```
rm autoscale profile <name>
```

Arguments

name

AutoScale profile name.

set autoscale profile

Set a AutoScale policy.

Synopsys

```
set autoscale profile <name> [-url <URL>] [-apiKey ] [-sharedSecret ]
```

Arguments

name

AutoScale profile name.

url

URL providing the service

apiKey

api key for authentication with service

sharedSecret

shared secret for authentication with service

show autoscale profile

Display the autoscale profile.

Synopsis

show autoscale profile [<name>]

Arguments

name

AutoScale profile name.

Outputs

type

The type of profile.

url

URL providing the service

apiKey

api key for authentication with service

sharedSecret

shared secret for authentication with service

stateflag

devno

count

Basic Commands

The entities on which you can perform NetScaler CLI operations:

- o configstatus
- o dbsMonitors
- o location
- o locationData
- o locationFile
- o locationParameter
- o nstrace
- o reporting
- o server
- o service
- o serviceGroup
- o serviceGroupMember
- o servicegroupbindings
- o svcbindings
- o uiinternal
- o vserver

configstatus

The following operations can be performed on "configstatus":

show configstatus

Display status of packet engines.

Synopsys

show configstatus

Outputs

consistent

State of packet engines.

culpritCore

Culprit core id.

core

Core id.

culpritCoreConfString

coreConfString

devno

count

stateflag

Example

```
show configstatus
```

dbMonitors

The following operations can be performed on "dbMonitors":

restart dbMonitors

Immediately send DNS queries to resolve the domain names of all the domain-based servers configured on the NetScaler appliance.

Synopsys

restart dbMonitors

Example

```
restart dbMonitors
```

location

The following operations can be performed on "location":

[add](#) | [rm](#) | [show](#)

add location

Creates a custom location entry on the NetScaler appliance. Custom locations can be used instead of a static location database if the number of locations you need does not exceed 500. Custom locations can also be used to override incorrect entries in the static database, because the appliance searches the static database before it searches the static location database.

Synopsys

```
add location <IPfrom> <IPto> <preferredLocation> [-longitude <integer> [-latitude <integer>]]
```

Arguments

IPfrom

First IP address in the range, in dotted decimal notation.

IPto

Last IP address in the range, in dotted decimal notation.

preferredLocation

String of qualifiers, in dotted notation, describing the geographical location of the IP address range. Each qualifier is more specific than the one that precedes it, as in continent.country.region.city.isp.organization. For example, "NA.US.CA.San Jose.ATT.citrix".

Note: A qualifier that includes a dot (.) or space () must be enclosed in double quotation marks.

longitude

Numerical value, in degrees, specifying the longitude of the geographical location of the IP address-range.

Note: Longitude and latitude parameters are used for selecting a service with the static proximity GSLB method. If they are not specified, selection is based on the qualifiers specified for the location.

Minimum value: -180

Maximum value: 180

latitude

Numerical value, in degrees, specifying the latitude of the geographical location of the IP address-range.

Note: Longitude and latitude parameters are used for selecting a service with the static proximity GSLB method. If they are not specified, selection is based on the qualifiers specified for the location.

Minimum value: -90

Maximum value: 90

Example

```
Add location 192.168.100.1 192.168.100.100 *.us.ca.san jose
```

rm location

Removes a custom location entry from the NetScaler appliance.

Synopsys

rm location <IPfrom> <IPto>

Arguments

IPfrom

First IP address in the range, in dotted decimal notation.

IPto

Last IP address in the range, in dotted decimal notation.

Example

```
rm location 192.168.100.1 192.168.100.100
```

show location

Displays all the custom location entries configured on the NetScaler appliance, or just the entry for the specified IP address range.

Synopsys

show location [<IPfrom>]

Arguments

IPfrom

The qualifiers in dotted notation for the ipaddress. If this value is not specified, all custom entries are displayed.

Outputs

IPto

The end of the IP address range.

preferredLocation

The qualifiers in dotted notation for the ipaddress range.

q1label

Least specific location qualifier.

q2label

Location qualifier 2.

q3label

Location qualifier 3.

q4label

Location qualifier 4.

q5label

Location qualifier 5.

q6label

Most specific location qualifier.

longitude

Numerical value, in degrees, specifying the longitude of the geographical location of the IP address-range.

Note: Longitude and latitude parameters are used for selecting a service with the static proximity GSLB method. If they are not specified, selection is based on the qualifiers specified for the location.

latitude

Numerical value, in degrees, specifying the latitude of the geographical location of the IP address-range.

Note: Longitude and latitude parameters are used for selecting a service with the static proximity GSLB method. If they are not specified, selection is based on the qualifiers specified for the location.

devno**count****stateflag**

Example

```
show location
```

locationData

The following operations can be performed on "locationData":

clear locationData

Clears all location information, including custom and static database entries.

Synopsys

```
clear locationData
```

Example

```
clear locationdata
```

locationFile

The following operations can be performed on "locationFile":

[add](#) | [rm](#) | [show](#)

add locationFile

Loads the static location database from the specified file.

Synopsis

```
add locationFile <locationFile> [-format <format>]
```

Arguments

locationFile

Name of the location file, with or without absolute path. If the path is not included, the default path (/var/netScaler/locdb) is assumed. In a high availability setup, the static database must be stored in the same location on both NetScaler appliances.

format

Format of the location file. Required for the NetScaler appliance to identify how to read the location file.

Possible values: netscaler, ip-country, ip-country-isp, ip-country-region-city, ip-country-region-city-isp, geoip-country, geoip-region, geoip-city, geoip-country-org, geoip-country-isp, geoip-city-isp-org

Default value: netscaler

Example

```
add locationfile /var/nsmap/locationdb -format netscaler
```

rm locationFile

Removes the currently loaded static location database from the NetScaler appliance.

Synopsis

```
rm locationFile
```

Example

```
rm locationfile
```

show locationFile

Displays the name, including the absolute path, and format of the location file currently loaded on the NetScaler appliance.

Synopsis

```
show locationFile
```

Outputs

locationFile

The name of the location file.

format

The format of the location file.

Example

```
show locationfile
```

locationParameter

The following operations can be performed on "locationParameter":

[set](#) | [unset](#) | [show](#)

set locationParameter

Sets the location parameters used for static-proximity based global server load balancing. Location parameters include up to six qualifiers and a context that specifies how the qualifiers must be interpreted. Each qualifier specifies the location of an IP address range and is more specific than the one that precedes it, as in continent.country.region.city.isp.organization. For example, "NA.US.CA.San Jose.ATT.citrix". Note: A qualifier that includes a dot (.) or space () must be enclosed in double quotation marks.

Synopsys

```
set locationParameter [-context ( geographic | custom )] [-q1label <string>] [-q2label <string>] [-q3label <string>] [-q4label <string>] [-q5label <string>] [-q6label <string>]
```

Arguments

context

Context for describing locations. In geographic context, qualifier labels are assigned by default in the following sequence: Continent.Country.Region.City.ISP.Organization. In custom context, the qualifiers labels can have any meaning that you designate.

Possible values: geographic, custom

q1label

Label specifying the meaning of the first qualifier. Can be specified for custom context only.

q2label

Label specifying the meaning of the second qualifier. Can be specified for custom context only.

q3label

Label specifying the meaning of the third qualifier. Can be specified for custom context only.

q4label

Label specifying the meaning of the fourth qualifier. Can be specified for custom context only.

q5label

Label specifying the meaning of the fifth qualifier. Can be specified for custom context only.

q6label

Label specifying the meaning of the sixth qualifier. Can be specified for custom context only.

Example

```
set locationparameter -context    custom
```

unset locationParameter

Use this command to remove locationParameter settings.Refer to the set locationParameter command for meanings of the arguments.

Synopsys

```
unset locationParameter [-context] [-q1label] [-q2label] [-q3label] [-q4label] [-q5label] [-q6label]
```

show locationParameter

Displays current values for the location parameters, which are used for static-proximity based load balancing.

Synopsys

show locationParameter

Outputs

context

The context in which a static proximity decision must be made.

q1label

The label for the 1st qualifier. These qualifier labels characterize the locations mapped with the IP addresses used to make static proximity decisions.

q2label

Label specifying the meaning of the second qualifier. Can be specified for custom context only.

q3label

Label specifying the meaning of the third qualifier. Can be specified for custom context only.

q4label

Label specifying the meaning of the fourth qualifier. Can be specified for custom context only.

q5label

Label specifying the meaning of the fifth qualifier. Can be specified for custom context only.

q6label

Label specifying the meaning of the sixth qualifier. Can be specified for custom context only.

locationFile

Currently loaded location database file.

format

custom

Number of configured custom locations.

static

Number of configured locations in the database file (static locations).

lines

Number of lines in the database files

errors

Number of errors encountered while reading the database file.

warnings

Number of warnings encountered while reading the database file.

entries

Number of successfully added entries.

flags

Information needed for display. This argument passes information from the kernel to the user space.

status

This argument displays when the status (success or failure) of database loading.

DatabaseMode

This argument displays the database mode.

flushing

This argument displays the state of flushing.

loading

This argument displays the state of loading.

Example

```
show locationparameter
```

nstrace

The following operations can be performed on "nstrace":

[start](#) | [stop](#) | [dump](#) | [show](#)

start nstrace

Start NetScaler packet capture tool.

Synopsis

```
start nstrace [-nf <positive_integer>] [-time <positive_integer>] [-size <positive_integer>] [-mode <mode> ...] [-tcpdump ( ENABLED | DISABLED )] [-perNIC ( ENABLED | DISABLED )] [-fileName <string>] [-fileId <string>] [-filter <expression>] [-link ( ENABLED | DISABLED )] [-nodes <positive_integer> ...] [-doruntimemerge ( ENABLED | DISABLED )] [-doruntimecleanup ( ENABLED | DISABLED )] [-traceBuffers <positive_integer>] [-skipRPC ( ENABLED | DISABLED )] [-inMemoryTrace ( ENABLED | DISABLED )]
```

Arguments

nf

Number of files to be generated in cycle.

Default value: 24

Minimum value: 1

Maximum value: 100

time

Time per file (sec).

Default value: 3600

Minimum value: 1

size

Size of the captured data. Set 0 for full packet trace.

Default value: 164

Minimum value: 0

Maximum value: 1514

mode

Capturing mode for trace. Mode can be any of the following values or combination of these values:

RX Received packets before NIC pipelining (Filter does not work when RX capturing mode is ON)

NEW_RX Received packets after NIC pipelining

TX Transmitted packets

TXB Packets buffered for transmission

IPV6 Translated IPv6 packets

C2C Capture C2C message

NS_FR_TX TX/TXB packets are not captured in flow receiver.

Default mode: NEW_RX TXB

Default value: DEFAULT_MODE

tcpdump

Trace is captured in TCPDUMP(.pcap) format. Default capture format is NSTRACE(.cap).

Possible values: ENABLED, DISABLED

Default value: DISABLED

perNIC

Use separate trace files for each interface. Works only with tcpdump format.

Possible values: ENABLED, DISABLED

Default value: DISABLED

fileName

Name of the trace file.

fileId

ID for the trace file name for uniqueness. Should be used only with -name option.

filter

Filter expression for nstrace. Maximum length of filter is 255 and it can be of following format:

<expression> [<relop> <expression>]

<relop> = (&& | ||)

nstrace supports two types of filter expressions:

Classic Expressions:

<expression> = the expression string in the format:

<qualifier> <operator> <qualifier-value>

<qualifier> = SOURCEIP.

<qualifier-value> = A valid IP address

<qualifier> = SOURCEPORT.

<qualifier-value> = A valid port number.

<qualifier> = DESTIP.

<qualifier-value> = A valid IP address.

<qualifier> = DESTPORT.

<qualifier-value> = A valid port number.

<qualifier> = IP.

<qualifier-value> = A valid IP address.

<qualifier> = PORT.

<qualifier-value> = A valid port number.

<qualifier> = SVCNAME.

<qualifier-value> = The name of a service.

<qualifier> = VSVRNAME.

<qualifier-value> = The name of a vserver.

<qualifier> = CONNID

<qualifier-value> = A valid PCB dev number.

<qualifier> = VLAN

<qualifier-value> = A valid VLAN ID.

<qualifier> = INTF

<qualifier-value> = A valid interface id in the form of x/y

(n/x/y in case of cluster interface).

<operator> = (== | eq | != | neq | > | gt

| < | lt | >= | ge | <= | le | BETWEEN)

eg: start nstrace -filter "SOURCEIP == 10.102.34.201 || (SVCNAME != s1 && SOURCEPORT > 80)"

The filter expression should be given in double quotes.

Default Expressions:

<expression> =:

CONNECTION.<qualifier>.<qualifier-method>.<qualifier-value>

<qualifier> = SRCIP

<qualifier-method> = [EQ | NE]

<qualifier-value> = A valid IPv4 address.

example = CONNECTION.SRCIP.EQ(127.0.0.1)

<qualifier> = DSTIP

<qualifier-method> = [EQ | NE]

<qualifier-value> = A valid IPv4 address.

example = CONNECTION.DSTIP.EQ(127.0.0.1)

<qualifier> = IP

<qualifier-method> = [EQ | NE]

<qualifier-value> = A valid IPv4 address.

example = CONNECTION.IP.EQ(127.0.0.1)

<qualifier> = SRCIPv6

<qualifier-method> = [EQ | NE]

<qualifier-value> = A valid IPv6 address.

example = CONNECTION.SRCIPv6.EQ(2001:db8:0:0:1::1)

<qualifier> = DSTIPv6

<qualifier-method> = [EQ | NE]

<qualifier-value> = A valid IPv6 address.

example = CONNECTION.DSTIPv6.EQ(2001:db8:0:0:1::1)

<qualifier> = IPv6

<qualifier-method> = [EQ | NE]

<qualifier-value> = A valid IPv6 address.

example = CONNECTION.IPv6.EQ(2001:db8:0:0:1::1)

<qualifier> = SRCPORT

<qualifier-method> = [EQ | NE | GT | GE | LT | LE
| BETWEEN]

<qualifier-value> = A valid port number.

example = CONNECTION.SRCPORT.EQ(80)

<qualifier> = DSTPORT

<qualifier-method> = [EQ | NE | GT | GE | LT | LE
| BETWEEN]

<qualifier-value> = A valid port number.

example = CONNECTION.DSTPORT.EQ(80)

<qualifier> = PORT

<qualifier-method> = [EQ | NE | GT | GE | LT | LE
| BETWEEN]

<qualifier-value> = A valid port number.

example = CONNECTION.PORT.EQ(80)

<qualifier> = VLANID

<qualifier-method> = [EQ | NE | GT | GE | LT | LE
| BETWEEN]

<qualifier-value> = A valid VLAN ID.

example = CONNECTION.VLANID.EQ(0)

<qualifier> = CONNID

<qualifier-method> = [EQ | NE | GT | GE | LT | LE
| BETWEEN]

<qualifier-value> = A valid PCB dev number.

example = CONNECTION.CONNID.EQ(0)

<qualifier> = PPEID

<qualifier-method> = [EQ | NE | GT | GE | LT | LE
| BETWEEN]

<qualifier-value> = A valid core ID.

example = CONNECTION.PPEID.EQ(0)

<qualifier> = SVCNAME

<qualifier-method> = [EQ | NE | CONTAINS | STARTSWITH
| ENDSWITH]

<qualifier-value> = A valid text string.

example = CONNECTION.SVCNAME.EQ("name")

<qualifier> = LB_VSERVER.NAME

<qualifier-method> = [EQ | NE | CONTAINS | STARTSWITH
| ENDSWITH]

<qualifier-value> = LB vserver name.

example = CONNECTION.LB_VSERVER.NAME.EQ("name")

<qualifier> = CS_VSERVER.NAME

<qualifier-method> = [EQ | NE | CONTAINS | STARTSWITH
| ENDSWITH]

<qualifier-value> = CS vserver name.

example = CONNECTION.CS_VSERVER.NAME.EQ("name")

<qualifier> = INTF

<qualifier-method> = [EQ | NE]

<qualifier-value> = A valid interface id in the
form of x/y.

example = CONNECTION.INTF.EQ("x/y")

<qualifier> = SERVICE_TYPE

<qualifier-method> = [EQ | NE]

<qualifier-value> = (SVC_HTTP | FTP | TCP | UDP | SSL |
SSL_BRIDGE | SSL_TCP | NNTP | RPCSVR | RPCSVRS |
RPCCLNT | SVC_DNS | ADNS | SNMP | RTSP | DHCPRA | ANY |
MONITOR | MONITOR_UDP | MONITOR_PING | SIP_UDP |
SVC_MYSQL | SVC_MSSQL | SERVICE_UNKNOWN)

example = CONNECTION.SERVICE_TYPE.EQ(ANY)

<qualifier> = TRAFFIC_DOMAIN_ID

<qualifier-method> = [EQ | NE | GT | GE | LT | LE
| BETWEEN]

<qualifier-value> = A valid traffic domain ID.

example = CONNECTION.TRAFFIC_DOMAIN_ID.EQ(0)

eg: start nstrace -filter "CONNECTION.SRCIP.EQ(127.0.0.1) || (CONNECTION.SVCNAME.NE("s1") &&
CONNECTION.SRCPORT.EQ(80))"

The filter expression should be given in double quotes.

common use cases:

Trace capturing full sized traffic from/to ip 10.102.44.111, excluding loopback traffic

start nstrace -size 0 -filter "CONNECTION.IP.NE(127.0.0.1) && CONNECTION.IP.EQ(10.102.44.111)"

Trace capturing all traffic to (terminating at) port 80 or 443

```
start nstrace -size 0 -filter "CONNECTION.DSTPORT.EQ(443) || CONNECTION.DSTPORT.EQ(80)"
```

Trace capturing all backend traffic specific to service service1 along with corresponding client side traffic

```
start nstrace -size 0 -filter "CONNECTION.SVCNAME.EQ("service1")" -link ENABLED
```

Trace capturing all traffic through NS interface 1/1

```
start nstrace -filter "CONNECTION.INTF.EQ("1/1")"
```

Trace capturing all traffic specific through vlan 2

```
start nstrace -filter "CONNECTION.VLANID.EQ(2)"
```

Trace capturing all frontend (client side) traffic specific to lb vserver vserver1 along with corresponding server side traffic

```
start nstrace -size 0 -filter "CONNECTION.LB_VSERVER.NAME.EQ("vserver1")" -link ENABLED
```

link

Includes filtered connection's peer traffic.

Possible values: ENABLED, DISABLED

Default value: DISABLED

nodes

Nodes on which tracing is started.

Minimum value: 0

Maximum value: 32

doruntime merge

Enable or disable runtime merge.

Possible values: ENABLED, DISABLED

Default value: ENABLED

doruntime cleanup

Enable or disable runtime temp file cleanup

Possible values: ENABLED, DISABLED

Default value: ENABLED

traceBuffers

Number of 16KB trace buffers

Default value: 5000

Minimum value: 1000

skipRPC

skip RPC packets

Possible values: ENABLED, DISABLED

Default value: DISABLED

inMemoryTrace

Logs packets in appliance's memory and dumps the trace file on stopping the nstrace operation

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
start nstrace -time 10
```

stop nstrace

Stop running NetScaler packet capture tool.

Synopsys

```
stop nstrace
```

Example

```
stop nstrace
```

dump nstrace

dump records from trace buffers to file.

Synopsys

```
dump nstrace -fileName <string>
```

Arguments

fileName

Name of the trace file.

Example

```
dump nstrace
```

show nstrace

Display nstrace parameters set through 'start nstrace' command.

Synopsys

```
show nstrace
```

Outputs

state

Current running state of trace.

scope

Scope of started trace, local or cluster level.

traceLocation

Directory where current trace files are saved.

nf

Number of files to be generated in cycle.

time

Time per file (sec).

size

Size of the captured data. Set 0 for full packet trace.

mode

Capturing mode for trace. Mode can be any of the following values or combination of these values:

RX Received packets before NIC pipelining (Filter does not work when RX capturing mode is ON)

NEW_RX Received packets after NIC pipelining

TX Transmitted packets

TXB Packets buffered for transmission

IPV6 Translated IPv6 packets

C2C Capture C2C message

NS_FR_TX TX/TXB packets are not captured in flow receiver.

Default mode: NEW_RX TXB

tcpdump

Trace is captured in TCPDUMP(.pcap) format. Default capture format is NSTRACE(.cap).

perNIC

Use separate trace files for each interface. Works only with tcpdump format.

fileName

Name of the trace file.

fileId

ID for the trace file name for uniqueness. Should be used only with -name option.

filter

Filter expression for nstrace. Maximum length of filter is 255 and it can be of following format:

<expression> [<relop> <expression>]

<relop> = (&& | ||)

nstrace supports two types of filter expressions:

Classic Expressions:

<expression> = the expression string in the format:

<qualifier> <operator> <qualifier-value>

<qualifier> = SOURCEIP.

<qualifier-value> = A valid IP address

<qualifier> = SOURCEPORT.

<qualifier-value> = A valid port number.

<qualifier> = DESTIP.

<qualifier-value> = A valid IP address.

<qualifier> = DESTPORT.

<qualifier-value> = A valid port number.

<qualifier> = IP.

<qualifier-value> = A valid IP address.

<qualifier> = PORT.

<qualifier-value> = A valid port number.

<qualifier> = SVCNAME.

<qualifier-value> = The name of a service.

<qualifier> = VSVRNAME.

<qualifier-value> = The name of a vserver.

<qualifier> = CONNID

<qualifier-value> = A valid PCB dev number.

<qualifier> = VLAN

<qualifier-value> = A valid VLAN ID.

<qualifier> = INTF

<qualifier-value> = A valid interface id in the form of x/y

(n/x/y in case of cluster interface).

<operator> = (== | eq | != | neq | > | gt

| < | lt | >= | ge | <= | le | BETWEEN)

eg: start nstrace -filter "SOURCEIP == 10.102.34.201 || (SVCNAME != s1 && SOURCEPORT > 80)"

The filter expression should be given in double quotes.

Default Expressions:

<expression> =:

CONNECTION.<qualifier>.<qualifier-method>.(<qualifier-value>)

<qualifier> = SRCIP

<qualifier-method> = [EQ | NE]

<qualifier-value> = A valid IPv4 address.

example = CONNECTION.SRCIP.EQ(127.0.0.1)

<qualifier> = DSTIP

<qualifier-method> = [EQ | NE]

<qualifier-value> = A valid IPv4 address.

example = CONNECTION.DSTIP.EQ(127.0.0.1)

<qualifier> = IP

<qualifier-method> = [EQ | NE]

<qualifier-value> = A valid IPv4 address.

example = CONNECTION.IP.EQ(127.0.0.1)

<qualifier> = SRCIPv6

<qualifier-method> = [EQ | NE]

<qualifier-value> = A valid IPv6 address.

example = CONNECTION.SRCIPv6.EQ(2001:db8:0:0:1::1)

<qualifier> = DSTIPv6

<qualifier-method> = [EQ | NE]

<qualifier-value> = A valid IPv6 address.

example = CONNECTION.DSTIPv6.EQ(2001:db8:0:0:1::1)

<qualifier> = IPv6

<qualifier-method> = [EQ | NE]

<qualifier-value> = A valid IPv6 address.

example = CONNECTION.IPv6.EQ(2001:db8:0:0:1::1)

<qualifier> = SRCPORT

<qualifier-method> = [EQ | NE | GT | GE | LT | LE
| BETWEEN]

<qualifier-value> = A valid port number.

example = CONNECTION.SRCPORT.EQ(80)

<qualifier> = DSTPORT

<qualifier-method> = [EQ | NE | GT | GE | LT | LE
| BETWEEN]

<qualifier-value> = A valid port number.

example = CONNECTION.DSTPORT.EQ(80)

<qualifier> = PORT

<qualifier-method> = [EQ | NE | GT | GE | LT | LE
| BETWEEN]

<qualifier-value> = A valid port number.

example = CONNECTION.PORT.EQ(80)

<qualifier> = VLANID

<qualifier-method> = [EQ | NE | GT | GE | LT | LE
| BETWEEN]

<qualifier-value> = A valid VLAN ID.

example = CONNECTION.VLANID.EQ(0)

<qualifier> = CONNID

<qualifier-method> = [EQ | NE | GT | GE | LT | LE
| BETWEEN]

<qualifier-value> = A valid PCB dev number.

example = CONNECTION.CONNID.EQ(0)

<qualifier> = PPEID

<qualifier-method> = [EQ | NE | GT | GE | LT | LE
| BETWEEN]

<qualifier-value> = A valid core ID.

example = CONNECTION.PPEID.EQ(0)

<qualifier> = SVCNAME

<qualifier-method> = [EQ | NE | CONTAINS | STARTSWITH
| ENDSWITH]

<qualifier-value> = A valid text string.

example = CONNECTION.SVCNAME.EQ("name")

<qualifier> = LB_VSERVER.NAME

<qualifier-method> = [EQ | NE | CONTAINS | STARTSWITH
| ENDSWITH]

<qualifier-value> = LB vserver name.

example = CONNECTION.LB_VSERVER.NAME.EQ("name")

<qualifier> = CS_VSERVER.NAME

<qualifier-method> = [EQ | NE | CONTAINS | STARTSWITH
| ENDSWITH]

<qualifier-value> = CS vserver name.

example = CONNECTION.CS_VSERVER.NAME.EQ("name")

<qualifier> = INTF

<qualifier-method> = [EQ | NE]

<qualifier-value> = A valid interface id in the
form of x/y.

example = CONNECTION.INTF.EQ("x/y")

<qualifier> = SERVICE_TYPE

<qualifier-method> = [EQ | NE]

<qualifier-value> = (SVC_HTTP | FTP | TCP | UDP | SSL |
SSL_BRIDGE | SSL_TCP | NNTP | RPCSVR | RPCSVRS |
RPCCLNT | SVC_DNS | ADNS | SNMP | RTSP | DHCPRA | ANY |
MONITOR | MONITOR_UDP | MONITOR_PING | SIP_UDP |
SVC_MYSQL | SVC_MSSQL | SERVICE_UNKNOWN)

example = CONNECTION.SERVICE_TYPE.EQ(ANY)

<qualifier> = TRAFFIC_DOMAIN_ID

<qualifier-method> = [EQ | NE | GT | GE | LT | LE

| BETWEEN]

<qualifier-value> = A valid traffic domain ID.

example = CONNECTION.TRAFFIC_DOMAIN_ID.EQ(0)

eg: start nstrace -filter "CONNECTION.SRCIP.EQ(127.0.0.1) || (CONNECTION.SVCNAME.NE("s1") && CONNECTION.SRCPORT.EQ(80))"

The filter expression should be given in double quotes.

common use cases:

Trace capturing full sized traffic from/to ip 10.102.44.111, excluding loopback traffic

start nstrace -size 0 -filter "CONNECTION.IP.NE(127.0.0.1) && CONNECTION.IP.EQ(10.102.44.111)"

Trace capturing all traffic to (terminating at) port 80 or 443

start nstrace -size 0 -filter "CONNECTION.DSTPORT.EQ(443) || CONNECTION.DSTPORT.EQ(80)"

Trace capturing all backend traffic specific to service service1 along with corresponding client side traffic

start nstrace -size 0 -filter "CONNECTION.SVCNAME.EQ("service1")" -link ENABLED

Trace capturing all traffic through NS interface 1/1

start nstrace -filter "CONNECTION.INTF.EQ("1/1")"

Trace capturing all traffic specific through vlan 2

start nstrace -filter "CONNECTION.VLANID.EQ(2)"

Trace capturing all frontend (client side) traffic specific to lb vserver vserver1 along with corresponding server side traffic

start nstrace -size 0 -filter "CONNECTION.LB_VSERVER.NAME.EQ("vserver1")" -link ENABLED

link

Includes filtered connection's peer traffic.

nodes

Nodes on which tracing is started.

doruntimemerge

Enable or disable runtime merge.

doruntimecleanup

Enable or disable runtime temp file cleanup

traceBuffers

Number of 16KB trace buffers

skipRPC

skip RPC packets

inMemoryTrace

Logs packets in appliance's memory and dumps the trace file on stopping the nstrace operation

Example

show nstrace

reporting

The following operations can be performed on "reporting":

[enable](#) | [disable](#) | [show](#)

enable reporting

Enable the data collection for reporting module.

Synopsys

enable reporting

Example

```
enable reporting
```

disable reporting

Disable the data collection for reporting module.

Synopsys

disable reporting

Example

```
disable reporting
```

show reporting

show the state of data collection for reporting module.

Synopsys

show reporting

Outputs

state

The rule associated with the entity

Example

```
show reporting
```

server

The following operations can be performed on "server":

add | **rm** | **set** | **unset** | **enable** | **disable** | **show** | **rename**

add server

Creates a server entry on the NetScaler appliance. The NetScaler appliance supports two types of servers: IP address based servers and domain based servers.

Synopsys

```
add server <name>@ (<IPAddress>@ | (<domain>@ [-domainResolveRetry <integer>] [-IPv6Address ( YES | NO
)]) | (-translationIp <ip_addr> -translationMask <netmask>)) [-state ( ENABLED | DISABLED )] [-comment <string>] [-
td <positive_integer>]
```

Arguments

name

Name for the server.

Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

Can be changed after the name is created.

IPAddress

IPv4 or IPv6 address of the server. If you create an IP address based server, you can specify the name of the server, instead of its IP address, when creating a service. Note: If you do not create a server entry, the server IP address that you enter when you create a service becomes the name of the server.

domain

Domain name of the server. For a domain based configuration, you must create the server first.

translationIp

IP address used to transform the server's DNS-resolved IP address.

translationMask

The netmask of the translation ip

domainResolveRetry

Time, in seconds, for which the NetScaler appliance must wait, after DNS resolution fails, before sending the next DNS query to resolve the domain name.

Default value: 5

Minimum value: 5

Maximum value: 20939

state

Initial state of the server.

Possible values: ENABLED, DISABLED

Default value: ENABLED

IPv6Address

Support IPv6 addressing mode. If you configure a server with the IPv6 addressing mode, you cannot use the server in the IPv4 addressing mode.

Possible values: YES, NO

Default value: NO

comment

Any information about the server.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

Example

```
add server web_serv 10.102.27.150
```

 To add multiple servers you can use the following command:

rm server

Removes a server entry from the NetScaler appliance.

Synopsis

```
rm server <name>@ ...
```

Arguments

name

Name of the server entry to remove.

Example

```
rm server web_svr
```

 To remove the servers named serv1, serv2 and serv3 at once you can use

set server

Modifies the specified parameters of a server entry.

Synopsis

```
set server <name>@ [-IPAddress <ip_addr|ipv6_addr|*>@ | -domainResolveRetry <integer> | -translationIp <ip_addr> | -translationMask <netmask> | -domainResolveNow] [-comment <string>]
```

Arguments

name

Name of the server whose parameters you are configuring.

IPAddress

Name of the server whose parameters you are configuring.

domainResolveRetry

Time, in seconds, for which the NetScaler appliance must wait, after DNS resolution fails, before sending the next DNS query to resolve the domain name.

Default value: 5

Minimum value: 5

Maximum value: 20939

translationIp

IP address used to transform the server's DNS-resolved IP address.

translationMask

The netmask of the translation ip

domainResolveNow

Immediately send a DNS query to resolve the server's domain name.

comment

Any information about the server.

Example

```
set server http_svr -IPAddress 10.102.1.112 To set multiple servers IP addresses at once
```

unset server

Use this command to remove server settings. Refer to the set server command for meanings of the arguments.

Synopsis

```
unset server <name>@ -comment
```

enable server

Enables all services on the specified server.

Synopsis

```
enable server <name>@
```

Arguments

name

Name of the server to enable.

Example

```
enable server web_serv To enable all the services configured on servers named serv1, serv2
```

disable server

Disables all services on the server. When a server is disabled, all services on the server are disabled.

Synopsis

```
disable server <name>@ [<delay>] [-graceFul ( YES | NO )]
```

Arguments

name

Name of the server to disable.

delay

Time, in seconds, after which all the services configured on the server are disabled.

graceFul

Shut down gracefully, without accepting any new connections, and disabling each service when all of its connections are closed.

Possible values: YES, NO

Default value: NO

Example

```
disable server web_svr 30 To disable all the services configured on servers named serv1
```

show server

Displays the parameters of all the server entries on the appliance, or the parameters of the specified server entry.

Synopsys

```
show server [<name> | -internal]
```

Arguments

name

Name of the server for which to display parameters.

internal

Display names of the servers that have been created for internal use.

Outputs

IPAddress

The IP Address of server.

state

The State of the server.

domain

The domain name of the server.

domainResolveRetry

Time, in seconds, for which the NetScaler appliance must wait, after DNS resolution fails, before sending the next DNS query to resolve the domain name.

serviceName

The services attached to the server.

serviceGroupName

servicegroups bind to this server

translationIp

IP address used to transform the server's DNS-resolved IP address.

translationMask

The netmask of the translation ip

comment

Any information about the server.

stateflag

stateflag

serviceType

service type of the service.

serviceIPAddress

The IP address of the bound service

serviceIPstr

This field has been introduced to show the db services ip

port

port of the service.

svrState

The state of the bound service

stateChangeTimeSec

Time when last state change happened. Seconds part.

ticksSinceLastStateChange

Time in 10 millisecond ticks since the last state change.

IPv6Address

Support IPv6 addressing mode. If you configure a server with the IPv6 addressing mode, you cannot use the server in the IPv4 addressing mode.

svrcfgFlags

service flags to denote its a db enabled.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

autoScale

Auto scale option for a servicegroup

CustomServerID

A positive integer to identify the service. Used when the persistency type is set to Custom Server ID.

monThreshold

Minimum sum of weights of the monitors that are bound to this service. Used to determine whether to mark a service as UP or DOWN.

maxClient

Maximum number of simultaneous open connections for the service group.

maxReq

Maximum number of requests that can be sent on a persistent connection to the service group.

Note: Connection requests beyond this value are rejected.

maxBandwidth

Maximum bandwidth, in Kbps, allocated for all the services in the service group.

usip

Use the client's IP address as the source IP address when initiating a connection to the server. When creating a service, if you do not set this parameter, the service inherits the global Use Source IP setting (available in the enable ns mode and disable ns mode CLI commands, or in the System > Settings > Configure modes > Configure Modes dialog box). However, you can override this setting after you create the service.

CKA

Enable client keep-alive for the service group.

TCPB

Enable TCP buffering for the service group.

CMP

Enable compression for the specified service.

cltTimeout

Time, in seconds, after which to terminate an idle client connection.

svrTimeout

Time, in seconds, after which to terminate an idle server connection.

cipHeader

Name of the HTTP header whose value must be set to the IP address of the client. Used with the Client IP parameter. If client IP insertion is enabled, and the client IP header is not specified, the value of Client IP Header parameter or the value set by the set ns config command is used as client's IP header name.

cip

Before forwarding a request to the service, insert an HTTP header with the client's IPv4 or IPv6 address as its value. Used if the server needs the client's IP address for security, accounting, or other purposes, and setting the Use Source IP parameter is not a viable option.

cacheable

Use the transparent cache redirection virtual server to forward the request to the cache server.

sc

State of the SureConnect feature for the service group.

sp

Enable surge protection for the service group.

downStateFlush

Perform delayed clean-up of connections to all services in the service group.

appflowLog

Enable logging of AppFlow information for the specified service group.

boundTD

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

devno

count

Example

```
> show server web_svr1  Name:                web_svr1      State:ENABLED  IPAddress:    10.10:
```

rename server

Renames a server.

Synopsys

```
rename server <name>@ <newName>@
```

Arguments

name

Existing name of the server.

newName

New name for the server. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

Example

```
rename server s1 s1-new
```


service

The following operations can be performed on "service":

add | **rm** | **set** | **unset** | **bind** | **unbind** | **enable** | **disable** | **show** | **rename** | **stat**

add service

Creates a service on the NetScaler appliance. If the service is domain based, before you create the service, create the server entry by using the add server command. Then, in this command, specify the Server parameter.

Synopsis

```
add service <name>@ (<IP>@ | <serverName>@) <serviceType> <port> [-clearTextPort <port>] [-cacheType <cacheType>] [-maxClient <positive_integer>] [-healthMonitor ( YES | NO )] [-maxReq <positive_integer>] [-cacheable ( YES | NO )] [-cip ( ENABLED | DISABLED ) [-cipHeader>]] [-usip ( YES | NO )] [-pathMonitor ( YES | NO )] [-pathMonitorIndv ( YES | NO )] [-useproxyport ( YES | NO )] [-sc ( ON | OFF )] [-sp ( ON | OFF )] [-rtspSessionidRemap ( ON | OFF )] [-cltTimeout <secs>] [-svrTimeout <secs>] [-CustomServerID <string>] [-CKA ( YES | NO )] [-TCPB ( YES | NO )] [-CMP ( YES | NO )] [-maxBandwidth <positive_integer>] [-accessDown ( YES | NO )] [-monThreshold <positive_integer>] [-state ( ENABLED | DISABLED )] [-downStateFlush ( ENABLED | DISABLED )] [-tcpProfileName <string>] [-httpProfileName <string>] [-hashId <positive_integer>] [-comment <string>] [-appflowLog ( ENABLED | DISABLED )] [-netProfile <string>] [-td <positive_integer>] [-processLocal ( ENABLED | DISABLED )]
```

Arguments

name

Name for the service. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the service has been created.

IP

IP to assign to the service.

serverName

Name of the server that hosts the service.

serviceType

Protocol in which data is exchanged with the service.

Possible values: HTTP, FTP, TCP, UDP, SSL, SSL_BRIDGE, SSL_TCP, DTLS, NNTP, RPCSVR, DNS, ADNS, SNMP, RTSP, DHCPRA, ANY, SIP_UDP, DNS_TCP, ADNS_TCP, MYSQL, MSSQL, ORACLE, RADIUS, RDP, DIAMETER, SSL_DIAMETER, TFTP

port

Port number of the service.

clearTextPort

Port to which clear text data must be sent after the appliance decrypts incoming SSL traffic. Applicable to transparent SSL services.

Minimum value: 1

cacheType

Cache type supported by the cache server.

Possible values: TRANSPARENT, REVERSE, FORWARD

maxClient

Maximum number of simultaneous open connections to the service.

Minimum value: 0

Maximum value: 4294967294

healthMonitor

Monitor the health of this service. Available settings function as follows:

YES - Send probes to check the health of the service.

NO - Do not send probes to check the health of the service. With the NO option, the appliance shows the service as UP at all times.

Possible values: YES, NO

Default value: YES

maxReq

Maximum number of requests that can be sent on a persistent connection to the service.

Note: Connection requests beyond this value are rejected.

Minimum value: 0

Maximum value: 65535

cacheable

Use the transparent cache redirection virtual server to forward requests to the cache server.

Note: Do not specify this parameter if you set the Cache Type parameter.

Possible values: YES, NO

Default value: NO

cip

Before forwarding a request to the service, insert an HTTP header with the client's IPv4 or IPv6 address as its value. Used if the server needs the client's IP address for security, accounting, or other purposes, and setting the Use Source IP parameter is not a viable option.

Possible values: ENABLED, DISABLED

cipHeader

Name for the HTTP header whose value must be set to the IP address of the client. Used with the Client IP parameter. If you set the Client IP parameter, and you do not specify a name for the header, the appliance uses the header name specified for the global Client IP Header parameter (the cipHeader parameter in the set ns param CLI command or the Client IP Header parameter in the Configure HTTP Parameters dialog box at System > Settings > Change HTTP parameters). If the global Client IP Header parameter is not specified, the appliance inserts a header with the name "client-ip."

usip

Use the client's IP address as the source IP address when initiating a connection to the server. When creating a service, if you do not set this parameter, the service inherits the global Use Source IP setting (available in the enable ns mode and disable ns mode CLI commands, or in the System > Settings > Configure modes > Configure Modes dialog box). However, you can override this setting after you create the service.

Possible values: YES, NO

pathMonitor

Path monitoring for clustering

Possible values: YES, NO

pathMonitorIndv

Individual Path monitoring decisions

Possible values: YES, NO

useproxyport

Use the proxy port as the source port when initiating connections with the server. With the NO setting, the client-side connection port is used as the source port for the server-side connection.

Note: This parameter is available only when the Use Source IP (USIP) parameter is set to YES.

Possible values: YES, NO

sc

State of SureConnect for the service.

Possible values: ON, OFF

Default value: OFF

sp

Enable surge protection for the service.

Possible values: ON, OFF

rtspSessionidRemap

Enable RTSP session ID mapping for the service.

Possible values: ON, OFF

Default value: OFF

cltTimeout

Time, in seconds, after which to terminate an idle client connection.

Maximum value: 31536000

svrTimeout

Time, in seconds, after which to terminate an idle server connection.

Maximum value: 31536000

CustomServerID

Unique identifier for the service. Used when the persistency type for the virtual server is set to Custom Server ID.

Default value: "None"

CKA

Enable client keep-alive for the service.

Possible values: YES, NO

TCPB

Enable TCP buffering for the service.

Possible values: YES, NO

CMP

Enable compression for the service.

Possible values: YES, NO

maxBandwidth

Maximum bandwidth, in Kbps, allocated to the service.

Minimum value: 0

Maximum value: 4294967287

accessDown

Use Layer 2 mode to bridge the packets sent to this service if it is marked as DOWN. If the service is DOWN, and this parameter is disabled, the packets are dropped.

Possible values: YES, NO

Default value: NO

monThreshold

Minimum sum of weights of the monitors that are bound to this service. Used to determine whether to mark a service as UP or DOWN.

Minimum value: 0

Maximum value: 65535

state

Initial state of the service.

Possible values: ENABLED, DISABLED

Default value: ENABLED

downStateFlush

Flush all active transactions associated with a service whose state transitions from UP to DOWN. Do not enable this option for applications that must complete their transactions.

Possible values: ENABLED, DISABLED

Default value: ENABLED

tcpProfileName

Name of the TCP profile that contains TCP configuration settings for the service.

httpProfileName

Name of the HTTP profile that contains HTTP configuration settings for the service.

hashId

A numerical identifier that can be used by hash based load balancing methods. Must be unique for each service.

Minimum value: 1

comment

Any information about the service.

appflowLog

Enable logging of AppFlow information.

Possible values: ENABLED, DISABLED

Default value: ENABLED

netProfile

Network profile to use for the service.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

processLocal

By turning on this option packets destined to a service in a cluster will not under go any steering. Turn this option for single packet request response mode or when the upstream device is performing a proper RSS for connection based distribution.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
add service http_svc 10.102.1.112 http 80
```

 The below command adds the service web_svc1 fo:

rm service

Removes a service.

Synopsys

```
rm service <name>@
```

Arguments

name

Name of the service.

Example

```
rm service http_svc
```

 To remove services svc1, svc2 and svc3 in one go use the following c:

set service

Modifies the parameters of an existing service.

Synopsys

```
set service <name>@ [-IPAddress <ip_addr|ipv6_addr|*>@] [-maxClient <positive_integer>] [-maxReq  
<positive_integer>] [-cacheable ( YES | NO )] [-cip ( ENABLED | DISABLED ) [<cipHeader>]] [-usip ( YES | NO )] [-  
pathMonitor ( YES | NO )] [-pathMonitorIndv ( YES | NO )] [-useproxyport ( YES | NO )] [-sc ( ON | OFF )] [-sp ( ON |  
OFF )] [-rtspSessionidRemap ( ON | OFF )] [-healthMonitor ( YES | NO )] [-cltTimeout <secs>] [-svrTimeout <secs>]  
[-CustomServerID <string>] [-CKA ( YES | NO )] [-TCPB ( YES | NO )] [-CMP ( YES | NO )] [-maxBandwidth  
<positive_integer>] [-accessDown ( YES | NO )] [-monThreshold <positive_integer>] [-weight <positive_integer>  
<monitorName>] [-downStateFlush ( ENABLED | DISABLED )] [-tcpProfileName <string>] [-httpProfileName  
<string>] [-hashId <positive_integer>] [-comment <string>] [-appflowLog ( ENABLED | DISABLED )] [-netProfile  
<string>] [-processLocal ( ENABLED | DISABLED )]
```

Arguments

name

Name of the service for which to modify parameters.

IPAddress

The new IP address of the service.

maxClient

Maximum number of simultaneous open connections to the service.

Minimum value: 0

Maximum value: 4294967294

maxReq

Maximum number of requests that can be sent on a persistent connection to the service.

Note: Connection requests beyond this value are rejected.

Minimum value: 0

Maximum value: 65535

cacheable

Use the transparent cache redirection virtual server to forward requests to the cache server.

Note: Do not specify this parameter if you set the Cache Type parameter.

Possible values: YES, NO

Default value: NO

cip

Before forwarding a request to the service, insert an HTTP header with the client's IPv4 or IPv6 address as its value. Used if the server needs the client's IP address for security, accounting, or other purposes, and setting the Use Source IP parameter is not a viable option.

Possible values: ENABLED, DISABLED

cipHeader

Name for the HTTP header whose value must be set to the IP address of the client. Used with the Client IP parameter. If you set the Client IP parameter, and you do not specify a name for the header, the appliance uses the header name specified for the global Client IP Header parameter (the cipHeader parameter in the set ns param CLI command or the Client IP Header parameter in the Configure HTTP Parameters dialog box at System > Settings > Change HTTP parameters). If the global Client IP Header parameter is not specified, the appliance inserts a header with the name "client-ip."

usip

Use the client's IP address as the source IP address when initiating a connection to the server. When creating a service, if you do not set this parameter, the service inherits the global Use Source IP setting (available in the enable ns mode and disable ns mode CLI commands, or in the System > Settings > Configure modes > Configure Modes dialog box). However, you can override this setting after you create the service.

Possible values: YES, NO

pathMonitor

Path monitoring for clustering

Possible values: YES, NO

pathMonitorIndv

Individual Path monitoring decisions

Possible values: YES, NO

useproxyport

Use the proxy port as the source port when initiating connections with the server. With the NO setting, the client-side connection port is used as the source port for the server-side connection.

Note: This parameter is available only when the Use Source IP (USIP) parameter is set to YES.

Possible values: YES, NO

sc

State of SureConnect for the service.

Possible values: ON, OFF

Default value: OFF

sp

Enable surge protection for the service.

Possible values: ON, OFF

rtspSessionidRemap

Enable RTSP session ID mapping for the service.

Possible values: ON, OFF

Default value: OFF

healthMonitor

Monitor the health of this service. Available settings function as follows:

YES - Send probes to check the health of the service.

NO - Do not send probes to check the health of the service. With the NO option, the appliance shows the service as UP at all times.

Possible values: YES, NO

Default value: YES

cltTimeout

Time, in seconds, after which to terminate an idle client connection.

Maximum value: 31536000

svrTimeout

Time, in seconds, after which to terminate an idle server connection.

Maximum value: 31536000

CustomServerID

Unique identifier for the service. Used when the persistency type for the virtual server is set to Custom Server ID.

Default value: "None"

CKA

Enable client keep-alive for the service.

Possible values: YES, NO

TCPB

Enable TCP buffering for the service.

Possible values: YES, NO

CMP

Enable compression for the service.

Possible values: YES, NO

maxBandwidth

Maximum bandwidth, in Kbps, allocated to the service.

Minimum value: 0

Maximum value: 4294967287

accessDown

Use Layer 2 mode to bridge the packets sent to this service if it is marked as DOWN. If the service is DOWN, and this parameter is disabled, the packets are dropped.

Possible values: YES, NO

Default value: NO

monThreshold

Minimum sum of weights of the monitors that are bound to this service. Used to determine whether to mark a service as UP or DOWN.

Minimum value: 0

Maximum value: 65535

weight

Weight to assign to the monitor-service binding. When a monitor is UP, the weight assigned to its binding with the service determines how much the monitor contributes toward keeping the health of the service above the value configured for the Monitor Threshold parameter.

Minimum value: 1

Maximum value: 100

monitorName

Name of the monitor bound to the specified service.

downStateFlush

Flush all active transactions associated with a service whose state transitions from UP to DOWN. Do not enable this option for applications that must complete their transactions.

Possible values: ENABLED, DISABLED

Default value: ENABLED

tcpProfileName

Name of the TCP profile that contains TCP configuration settings for the service.

httpProfileName

Name of the HTTP profile that contains HTTP configuration settings for the service.

hashId

A numerical identifier that can be used by hash based load balancing methods. Must be unique for each service.

Minimum value: 1

comment

Any information about the service.

appflowLog

Enable logging of AppFlow information.

Possible values: ENABLED, DISABLED

Default value: ENABLED

netProfile

Network profile to use for the service.

processLocal

By turning on this option packets destined to a service in a cluster will not under go any steering. Turn this option for single packet request response mode or when the upstream device is performing a proper RSS for connection based distribution.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
set service http_svc -maxClient 100 The following command sets IP address 10.102.27.53 for
```

unset service

Removes the parameter settings of the specified service. Attributes for which a default value is available revert to their default values..Refer to the set service command for meanings of the arguments.

Synopsys

```
unset service <name>@ [-maxClient] [-maxReq] [-cacheable] [-cip] [-usip] [-pathMonitor] [-pathMonitorIndv] [-useproxyport] [-sc] [-sp] [-rtspSessionidRemap] [-CustomServerID] [-CKA] [-TCPB] [-CMP] [-maxBandwidth] [-accessDown] [-monThreshold] [-cltTimeout] [-riseApbrStatsMsgCode] [-svrTimeout] [-tcpProfileName] [-httpProfileName] [-hashId] [-appflowLog] [-netProfile] [-processLocal] [-cipHeader] [-healthMonitor] [-downStateFlush] [-comment]
```

Example

```
unset service http_svc -maxClient To unset maxclients for services svc1, svc2 and svc3, 1
```

bind service

Binds a policy or a monitor to a service.

Synopsys

```
bind service <name>@ (-policyName <string> | (-monitorName <string>@ [-monState ( ENABLED | DISABLED )] [-weight <positive_integer>] [-passive]))
```

Arguments

name

Name of the service to which to bind a policy or monitor.

policyName

Name of the policy to bind to the service.

monitorName

Name of the monitor to bind to the service.

monState

Initial state of the service.

Possible values: ENABLED, DISABLED

Default value: ENABLED

weight

Weight to assign to the monitor-service binding. When a monitor is UP, the weight assigned to its binding with the service determines how much the monitor contributes toward keeping the health of the service above the value configured for the Monitor Threshold parameter.

Default value: 1

Minimum value: 1

Maximum value: 100

passive

Indicates if load monitor is passive. A passive load monitor does not remove service from LB decision when threshold is breached.

Example

```
bind service svc1 -policyName poll1 To bind svc1, svc2 and svc3 to the policy poll1 you can
```

unbind service

Unbinds a policy or monitor from the specified service.

Synopsis

```
unbind service <name>@ (-policyName <string> | -monitorName <string>@)
```

Arguments

name

Name of the service from which to unbind a policy or monitor.

policyName

Name of the policy to unbind.

monitorName

Name of the monitor assigned to the service.

Example

```
unbind service http_svc -policyName poll1 To unbind a policy called poll1 on services svc1
```

enable service

Enables a service.

Synopsys

enable service <name>@

Arguments

name

Name of the service.

Example

`enable service http_svc` To enable svc1, svc2 and svc3 in one go use the following command

disable service

Disables a service.

Synopsys

disable service <name>@ [<delay>] [-graceFul (YES | NO)]

Arguments

name

Name of the service.

delay

Time, in seconds, allocated to the NetScaler appliance for a graceful shutdown of the service. During this period, new requests are sent to the service only for clients who already have persistent sessions on the appliance. Requests from new clients are load balanced among other available services. After the delay time expires, no requests are sent to the service, and the service is marked as unavailable (OUT OF SERVICE).

graceFul

Shut down gracefully, not accepting any new connections, and disabling the service when all of its connections are closed.

Possible values: YES, NO

Default value: NO

Example

`disable service http_svc 10` To disable svc1, svc2 and svc3 in one go use the following command

show service

Displays a list of all services configured on the NetScaler appliance, or the configuration details of the specified service.

Synopsys

show service [<name> | -all | -internal] show service bindings - alias for 'show svcbindings'

Arguments

name

Name of the service for which to display configuration details.

all

Display both user-configured and dynamically learned services.

internal

Display only dynamically learned services.

Outputs

numOfconnections

This will tell the number of client side connections are still open.

serverName

The name of the server for which a service has created.

policyName

The name of the polycyname for which this service is bound

serviceType

The type of service

serviceConfType

The configuration type of the service

serviceConfType

The configuration type of the service

serviceConfType2

The configuration type of the service (Internal/Dynamic/Configured).

port

Port number of the service.

value

SSL status.

clearTextPort

The clear-text port number where clear-text data is sent. Used with SSL offload service

gslb

The GSLB option for the corresponding virtual server.

cacheType

Cache type supported by the cache server.

maxClient

Maximum number of simultaneous open connections to the service.

maxReq

Maximum number of requests that can be sent on a persistent connection to the service.

Note: Connection requests beyond this value are rejected.

cacheable

Use the transparent cache redirection virtual server to forward requests to the cache server.

Note: Do not specify this parameter if you set the Cache Type parameter.

cip

Before forwarding a request to the service, insert an HTTP header with the client's IPv4 or IPv6 address as its value. Used if the server needs the client's IP address for security, accounting, or other purposes, and setting the Use Source IP parameter is not a viable option.

cipHeader

The client IP header.

usip

The use of client's IP Address option.

pathMonitor

Path monitoring for clustering

pathMonitorIndv

Individual Path monitoring for decisions.

useproxyport

The use of client's Port.

sc

The state of SureConnect for the service.

weight

The weight for the specified monitor.

state

Initial state of the service.

sp

Enable surge protection for the service.

rtspSessionidRemap

Enable RTSP session ID mapping for the service.

failedprobes

Number of the current failed monitoring probes.

cltTimeout

The idle time in seconds after which the client connection is terminated.

totalprobes

The total number of probs sent.

svrTimeout

The idle time in seconds after which the server connection is terminated.

totalfailedprobes

The total number of failed probs.

publicIP

public ip

publicPort

public port

CustomServerID

The identifier for the service. Used when the persistency type is set to Custom Server ID.

serverID

The identifier for the service. This is used when the persistency type is set to Custom Server ID.

CKA

Enable client keep-alive for the service.

TCPB

Enable TCP buffering for the service.

processLocal

By turning on this option packets destined to a service in a cluster will not under go any steering. Turn this option for single packet request response mode or when the upstream device is performing a proper RSS for connection based distribution.

CMP

Enable compression for the service.

maxBandwidth

The maximum bandwidth in kbps allowed for the service

accessDown

The option to allow access to disabled or down services. If enabled, all packets to the service are bridged; if disabled, they are dropped.

svrState

The state of the service

delay

The remaining time in seconds for the service to be disabled

IPAddress

The IP address of the server.

monitorName

The monitor Names.

monThreshold

Minimum sum of weights of the monitors that are bound to this service. Used to determine whether to mark a service as UP or DOWN.

monState

The running state of the monitor on this service.

monStatCode

The code indicating the monitor response.

lastresponse

The string form of monstatcode.

responseTime

Response time of this monitor.

riseApbrStatsMsgCode

The code indicating the rise apbr status.

riseApbrStatsMsgCode2

The code indicating other rise stats.

monStatParam1

First parameter for use with message code.

monStatParam2

Second parameter for use with message code.

monStatParam3

Third parameter for use with message code.

downStateFlush

Flush all active transactions associated with a service whose state transitions from UP to DOWN. Do not enable this option for applications that must complete their transactions.

stateChangeTimeSec

Time when last state change happened. Seconds part.

stateChangeTimemSec

Time at which last state change happened. Milliseconds part.

timeSinceLastStateChange

Time in milliseconds since the last state change.

ticksSinceLastStateChange

Time in 10 millisecond ticks since the last state change.

StateUpdateReason

Checks state update reason on the secondary node.

CIMonOwner

Tells the mon owner of the service.

CIMonView

Tells the view id of the monitoring owner.

tcpProfileName

Name of the TCP profile.

httpProfileName

Name of the HTTP profile that contains HTTP configuration settings for the service.

hashId

A numerical identifier that can be used by hash based load balancing methods. Must be unique for each service.

graceFul

Indicates graceful shutdown of the service. System will wait for all outstanding connections to this service to be closed before disabling the service.

comment

Comments associated with this service.

monitorTotalProbes

Total number of probes sent to monitor this service.

monitorTotalFailedProbes

Total number of failed probes

monitorCurrentFailedProbes

Total number of currently failed probes

stateflag

stateflag

healthMonitor

Monitor the health of this service. Available settings function as follows:

YES - Send probes to check the health of the service.

NO - Do not send probes to check the health of the service. With the NO option, the appliance shows the service as UP at all times.

appflowLog

Enable logging of AppFlow information.

netProfile

Network profile to use for the service.

svccfgFlags

Contains the information about config info like internal/configured service

serviceIPstr

This field has been introduced to show the db services ip

svcMonFlags

to store the flags of monitor bound to it

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

passive

Indicates if load monitor is passive. A passive load monitor does not remove service from LB decision when threshold is breached.

oracleServerVersion

Oracle server version

devno**count**

Example

The following is sample output of the `show service -all` command: 4 configured services: 1

rename service

Renames a service.

Synopsys

```
rename service <name>@ <newName>@
```

Arguments

name

Existing name of the service to be renamed.

newName

New name for the service. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

Example

```
rename service svcl svcnew
```

stat service

Displays statistics that have been collected for the specified service.

Synopsys

```
stat service [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full )]
```

Arguments

name

Name of the service.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Throughput (Mbps) (Throughput)

Number of bytes received or sent by this service (Mbps).

Average server TTFB (SvrTTFB)

Average TTFB between the NetScaler appliance and the server. TTFB is the time interval between sending the request packet to a service and receiving the first response from the service

IP address (IP)

The IP address on which the service is running.

Port (port)

The port on which the service is running.

Service type (Type)

The service type of this service. Possible values are ADNS, DNS, MYSQL, RTSP, SSL_DIAMETER, ADNS_TCP, DNS_TCP, NNTP, SIP_UDP, SSL_TCP, ANY, FTP, RADIUS, SNMP, TCP, DHCPRA, HTTP, RDP, SSL, TFTP, DIAMETER, MSSQL, RPCSVR, SSL_BRIDGE, UDP

State

Current state of the server. Possible values are UP, DOWN, UNKNOWN, OFS(Out of Service), TROFS (Transition Out of Service), TROFS_DOWN(Down When going Out of Service)

Requests (Req)

Total number of requests received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

Responses (Rsp)

Number of responses received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

Request bytes (Reqb)

Total number of request bytes received on this service or virtual server.

Response bytes (Rspb)

Number of response bytes received by this service or virtual server.

Current client connections (CIntConn)

Number of current client connections.

Requests in surge queue (SurgeQ)

Number of requests in the surge queue.

Current server connections (SvrConn)

Number of current connections to the actual servers behind the virtual server.

Current Server Est connections (SvrEstConn)

Number of server connections in ESTABLISHED state.

Connections in reuse pool (ReuseP)

Number of requests in the idle queue/reuse pool.

Maximum server connections (MaxConn)

Maximum open connections allowed on this service.

Current load on the service (Load)

Load on the service that is calculated from the bound load based monitor.

Current flags on the service (CurtFlags)

Current flags on the service for internal use in display handlers.

Service hits (Hits)

Number of times that the service has been provided.

ActvTrans

Number of active transactions handled by this service. (Including those in the surge queue.)

Active Transaction means number of transactions currently served by the server including those waiting in the SurgeQ

Total Packets rcvd (PktRx)

Total number of packets received by this service or virtual server.

Total Packets sent (PktTx)

Total number of packets sent.

serviceGroup

The following operations can be performed on "serviceGroup":

add | **rm** | **set** | **unset** | **bind** | **unbind** | **enable** | **disable** | **show** | **stat** | **rename**

add serviceGroup

Creates a service group. You can group similar services into a service group and use them as a single entity.

Synopsys

```
add serviceGroup <serviceName>@ <serviceType> [-cacheType <cacheType>] [-td <positive_integer>] [-maxClient <positive_integer>] [-maxReq <positive_integer>] [-cacheable ( YES | NO )] [-cip ( ENABLED | DISABLED )] [-cipHeader] [-usip ( YES | NO )] [-pathMonitor ( YES | NO )] [-pathMonitorIndv ( YES | NO )] [-useproxyport ( YES | NO )] [-healthMonitor ( YES | NO )] [-sc ( ON | OFF )] [-sp ( ON | OFF )] [-rtspSessionidRemap ( ON | OFF )] [-cltTimeout <secs>] [-svrTimeout <secs>] [-CKA ( YES | NO )] [-TCPB ( YES | NO )] [-CMP ( YES | NO )] [-maxBandwidth <positive_integer>] [-monThreshold <positive_integer>] [-state ( ENABLED | DISABLED )] [-downStateFlush ( ENABLED | DISABLED )] [-tcpProfileName <string>] [-httpProfileName <string>] [-comment <string>] [-appflowLog ( ENABLED | DISABLED )] [-netProfile <string>] [-autoScale <autoScale> -memberPort <port>]
```

Arguments

serviceName

Name of the service group. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Can be changed after the name is created.

serviceType

Protocol used to exchange data with the service.

Possible values: HTTP, FTP, TCP, UDP, SSL, SSL_BRIDGE, SSL_TCP, DTLS, NNTP, RPCSVR, DNS, ADNS, SNMP, RTSP, DHCPRA, ANY, SIP_UDP, DNS_TCP, ADNS_TCP, MYSQL, MSSQL, ORACLE, RADIUS, RDP, DIAMETER, SSL_DIAMETER, TFTP

cacheType

Cache type supported by the cache server.

Possible values: TRANSPARENT, REVERSE, FORWARD

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

maxClient

Maximum number of simultaneous open connections for the service group.

Minimum value: 0

Maximum value: 4294967294

maxReq

Maximum number of requests that can be sent on a persistent connection to the service group.

Note: Connection requests beyond this value are rejected.

Minimum value: 0

Maximum value: 65535

cacheable

Use the transparent cache redirection virtual server to forward the request to the cache server.

Note: Do not set this parameter if you set the Cache Type.

Possible values: YES, NO

Default value: NO

cip

Insert the Client IP header in requests forwarded to the service.

Possible values: ENABLED, DISABLED

cipHeader

Name of the HTTP header whose value must be set to the IP address of the client. Used with the Client IP parameter. If client IP insertion is enabled, and the client IP header is not specified, the value of Client IP Header parameter or the value set by the set ns config command is used as client's IP header name.

usip

Use client's IP address as the source IP address when initiating connection to the server. With the NO setting, which is the default, a mapped IP (MIP) address or subnet IP (SNIP) address is used as the source IP address to initiate server side connections.

Possible values: YES, NO

pathMonitor

Path monitoring for clustering

Possible values: YES, NO

pathMonitorIndv

Individual Path monitoring decisions.

Possible values: YES, NO

useproxyport

Use the proxy port as the source port when initiating connections with the server. With the NO setting, the client-side connection port is used as the source port for the server-side connection.

Note: This parameter is available only when the Use Source IP (USIP) parameter is set to YES.

Possible values: YES, NO

healthMonitor

Monitor the health of this service. Available settings function as follows:

YES - Send probes to check the health of the service.

NO - Do not send probes to check the health of the service. With the NO option, the appliance shows the service as UP at all times.

Possible values: YES, NO

Default value: YES

sc

State of the SureConnect feature for the service group.

Possible values: ON, OFF

Default value: OFF

sp

Enable surge protection for the service group.

Possible values: ON, OFF

Default value: OFF

rtspSessionidRemap

Enable RTSP session ID mapping for the service group.

Possible values: ON, OFF

Default value: OFF

cltTimeout

Time, in seconds, after which to terminate an idle client connection.

Maximum value: 31536000

svrTimeout

Time, in seconds, after which to terminate an idle server connection.

Maximum value: 31536000

CKA

Enable client keep-alive for the service group.

Possible values: YES, NO

TCPB

Enable TCP buffering for the service group.

Possible values: YES, NO

CMP

Enable compression for the specified service.

Possible values: YES, NO

maxBandwidth

Maximum bandwidth, in Kbps, allocated for all the services in the service group.

Minimum value: 0

Maximum value: 4294967287

monThreshold

Minimum sum of weights of the monitors that are bound to this service. Used to determine whether to mark a service as UP or DOWN.

Minimum value: 0

Maximum value: 65535

state

Initial state of the service group.

Possible values: ENABLED, DISABLED

Default value: ENABLED

downStateFlush

Flush all active transactions associated with all the services in the service group whose state transitions from UP to DOWN. Do not enable this option for applications that must complete their transactions.

Possible values: ENABLED, DISABLED

Default value: ENABLED

tcpProfileName

Name of the TCP profile that contains TCP configuration settings for the service group.

httpProfileName

Name of the HTTP profile that contains HTTP configuration settings for the service group.

comment

Any information about the service group.

appflowLog

Enable logging of AppFlow information for the specified service group.

Possible values: ENABLED, DISABLED

Default value: ENABLED

netProfile

Network profile for the service group.

autoScale

Auto scale option for a servicegroup

Possible values: DISABLED, DNS, POLICY

Default value: DISABLED

memberPort

member port

Example

```
add servicegroup http_svc_group http To add service groups sgrp1, sgrp2 and sgrp3 at one
```

rm serviceGroup

Removes a service group.

Synopsis

```
rm serviceGroup <serviceName>@
```

Arguments

serviceName

Name of the service group.

Example

```
rm servicegroup http_svc_group To remove multiple servicegroups at once, the following c
```

set serviceGroup

Modifies the specified parameters of a service group.

Synopsis

```
set serviceGroup <serviceName>@ [(<serverName>@ <port> [-weight <positive_integer>] [-CustomServerID <string>] [-hashId <positive_integer>]) | -maxClient <positive_integer> | -maxReq <positive_integer> | -cacheable ( YES | NO ) | -cip ( ENABLED | DISABLED ) | <cipHeader> | -usip ( YES | NO ) | -useproxyport ( YES | NO ) | -sc ( ON | OFF ) | -sp ( ON | OFF ) | -rtspSessionidRemap ( ON | OFF ) | -cltTimeout <secs> | -svrTimeout <secs> | -CKA ( YES | NO ) | -TCPB ( YES | NO ) | -CMP ( YES | NO ) | -maxBandwidth <positive_integer> | -monThreshold <positive_integer> | -downStateFlush ( ENABLED | DISABLED )] [-monitorName <string> -weight <positive_integer>] [-healthMonitor ( YES | NO )] [-pathMonitor ( YES | NO )] [-pathMonitorIndv ( YES | NO )] [-tcpProfileName <string>] [-httpProfileName <string>] [-comment <string>] [-appflowLog ( ENABLED | DISABLED )] [-netProfile <string>]
```

Arguments

serviceName

Name of the service group.

serverName

Name of the server to which to bind the service group.

port

Server port number.

weight

weight of the monitor that is bound to servicegroup.

Minimum value: 1

CustomServerID

The identifier for this IP:Port pair. Used when the persistency type is set to Custom Server ID.

Default value: "None"

hashId

The hash identifier for the service. This must be unique for each service. This parameter is used by hash based load balancing methods.

Minimum value: 1

monitorName

Name of the monitor bound to the service group. Used to assign a weight to the monitor.

maxClient

Maximum number of simultaneous open connections for the service group.

Minimum value: 0

Maximum value: 4294967294

maxReq

Maximum number of requests that can be sent on a persistent connection to the service group.

Note: Connection requests beyond this value are rejected.

Minimum value: 0

Maximum value: 65535

healthMonitor

Monitor the health of this service. Available settings function as follows:

YES - Send probes to check the health of the service.

NO - Do not send probes to check the health of the service. With the NO option, the appliance shows the service as UP at all times.

Possible values: YES, NO

Default value: YES

cacheable

Use the transparent cache redirection virtual server to forward the request to the cache server.

Note: Do not set this parameter if you set the Cache Type.

Possible values: YES, NO

Default value: NO

cip

Insert the Client IP header in requests forwarded to the service.

Possible values: ENABLED, DISABLED

cipHeader

CIP Header.

usip

Use client's IP address as the source IP address when initiating connection to the server. With the NO setting, which is the default, a mapped IP (MIP) address or subnet IP (SNIP) address is used as the source IP address to initiate server side connections.

Possible values: YES, NO

pathMonitor

Path monitoring for clustering

Possible values: YES, NO

pathMonitorIndv

Individual Path monitoring decisions.

Possible values: YES, NO

useproxyport

Use the proxy port as the source port when initiating connections with the server. With the NO setting, the client-side connection port is used as the source port for the server-side connection.

Note: This parameter is available only when the Use Source IP (USIP) parameter is set to YES.

Possible values: YES, NO

sc

State of the SureConnect feature for the service group.

Possible values: ON, OFF

Default value: OFF

sp

Enable surge protection for the service group.

Possible values: ON, OFF

Default value: OFF

rtspSessionidRemap

Enable RTSP session ID mapping for the service group.

Possible values: ON, OFF

Default value: OFF

cltTimeout

Time, in seconds, after which to terminate an idle client connection.

Maximum value: 31536000

svrTimeout

Time, in seconds, after which to terminate an idle server connection.

Maximum value: 31536000

CKA

Enable client keep-alive for the service group.

Possible values: YES, NO

TCPB

Enable TCP buffering for the service group.

Possible values: YES, NO

CMP

Enable compression for the specified service.

Possible values: YES, NO

maxBandwidth

Maximum bandwidth, in Kbps, allocated for all the services in the service group.

Minimum value: 0

Maximum value: 4294967287

monThreshold

Minimum sum of weights of the monitors that are bound to this service. Used to determine whether to mark a service as UP or DOWN.

Minimum value: 0

Maximum value: 65535

downStateFlush

Flush all active transactions associated with all the services in the service group whose state transitions from UP to DOWN. Do not enable this option for applications that must complete their transactions.

Possible values: ENABLED, DISABLED

Default value: ENABLED

tcpProfileName

Name of the TCP profile that contains TCP configuration settings for the service group.

httpProfileName

Name of the HTTP profile that contains HTTP configuration settings for the service group.

comment

Any information about the service group.

appflowLog

Enable logging of AppFlow information for the specified service group.

Possible values: ENABLED, DISABLED

Default value: ENABLED

netProfile

Network profile for the service group.

Example

```
set servicegroup http_svc_group -maxClient 100 To set the attribute maxclient for multip.
```

unset serviceGroup

Removes the attributes of the specified service group. Attributes for which a default value is available revert to their default values..Refer to the set serviceGroup command for meanings of the arguments.

Synopsis

```
unset serviceGroup <serviceName>@ [<serverName>@ <port> [-weight] [-CustomServerID] [-hashId] [-riseApbrStatsMsgCode]] [-maxClient] [-maxReq] [-cacheable] [-cip] [-usip] [-useproxyport] [-sc] [-sp] [-rtspSessionidRemap] [-cltTimeout] [-svrTimeout] [-CKA] [-TCPB] [-CMP] [-maxBandwidth] [-monThreshold] [-tcpProfileName] [-httpProfileName] [-appflowLog] [-netProfile] [-monitorName] [-weight] [-healthMonitor] [-cipHeader] [-pathMonitor] [-pathMonitorIndv] [-downStateFlush] [-comment]
```

Example

```
unset servicegroup http_svc_group -maxClient
```

bind serviceGroup

Binds a service to a service group.

Synopsis

```
bind serviceGroup <serviceName> ((<IP>@ <port>) | <serverName>@ | ((-monitorName <string>@ [-monState ( ENABLED | DISABLED )] [-passive]) | -CustomServerID <string> | -state ( ENABLED | DISABLED ) | -hashId <positive_integer> | )) [-weight <positive_integer>]
```

Arguments

serviceName

Name of the service group.

IP

IP address of the server that hosts the service. Mutually exclusive with the Server Name parameter.

serverName

Name of the server that hosts the service. Mutually exclusive with the IP address parameter.

port

Port number of the service. Each service must have a unique port number.

monitorName

The name of the service or a service group to which the monitor is to be bound.

monState

Administrative state assigned to the monitor and service group binding. If set to disabled, the service group is not monitored.

Possible values: ENABLED, DISABLED

Default value: ENABLED

passive

Indicates if load monitor is passive. A passive load monitor does not remove service from LB decision when threshold is breached.

weight**CustomServerID**

Unique service identifier. Used when the persistency type for the virtual server is set to Custom Server ID.

Default value: "None"

state

Initial state of the service after binding.

Possible values: ENABLED, DISABLED

Default value: ENABLED

hashId

Unique numerical identifier used by hash based load balancing methods to identify a service.

Minimum value: 1

Example

```
bind servicegroup http_svc_group 10.102.27.153 80 To bind multiple servers to a serviceg:
```

unbind serviceGroup

Unbinds a service or a monitor from a service group.

Synopsys

```
unbind serviceGroup <serviceGroupName> ((<IP>@ <port>) | <serverName>@ | -monitorName <string>@)
```

Arguments

serviceGroupName

Name of the service group.

IP

IP address of the server that hosts the service. Mutually exclusive with the Server Name parameter.

serverName

Name of the server that hosts the service. Mutually exclusive with the IP Address parameter.

port

Port number of the service.

monitorName

Name of the monitor to bind to the service group.

Example

```
unbind servicegroup http_svc_group 10.102.27.153 80 To unbind multiple servers following
```

enable serviceGroup

Enables a service group or a member of the service group.

Synopsis

```
enable serviceGroup <serviceName>@ [<serverName>@ <port>]
```

Arguments

serviceName

Name of the service group.

serverName

Name of the server that hosts the service.

port

Port number of the service to be enabled.

Example

```
enable servicegroup http_svc_group To enable multiple service groups at one go use the f
```

disable serviceGroup

Disables a service group or a member of a service group. To disable a service group, provide only the service group name. To disable only a member of a service group, in addition to the service group name, provide the name of the server that hosts the service, and the port number of the service.

Synopsis

```
disable serviceGroup <serviceName>@ [<serverName>@ <port>] [-delay <secs>] [-graceFul ( YES | NO )]
```

Arguments

serviceName

Name of the service group.

serverName

Name of the server that hosts the service.

port

Port number of the service.

delay

Time, in seconds, allocated for a shutdown of the services in the service group. During this period, new requests are sent to the service only for clients who already have persistent sessions on the appliance. Requests from new clients are load balanced among other available services. After the delay time expires, no requests are sent to the service, and the service is marked as unavailable (OUT OF SERVICE).

graceFul

Wait for all existing connections to the service to terminate before shutting down the service.

Possible values: YES, NO

Default value: NO

Example

```
disable servicegroup http_svc_group 10.102.27.153 80 -delay 10 To disable multiple servi
```

show serviceGroup

Displays the specified service group's binding information.

Synopsys

```
show serviceGroup [<serviceGroupName> | -includeMembers]
```

Arguments

serviceGroupName

Name of the service group.

includeMembers

Display the members of the listed service groups in addition to their settings. Can be specified when no service group name is provided in the command. In that case, the details displayed for each service group are identical to the details displayed when a service group name is provided, except that bound monitors are not displayed.

Outputs

numOfconnections

This will tell the number of client side connections are still open.

serviceType

Protocol used to exchange data with the service.

port

The port number of the service to be enabled.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

serviceConfType

serviceConfType

The configuration type of the service group.

value

SSL Status.

cacheType

Cache type supported by the cache server.

maxClient

Maximum number of simultaneous open connections for the service group.

maxReq

Maximum number of requests that can be sent on a persistent connection to the service group.

Note: Connection requests beyond this value are rejected.

cacheable

The state of cache on the service.

cip

Insert the Client IP header in requests forwarded to the service.

cipHeader

CIP Header.

usip

Use client's IP address as the source IP address when initiating connection to the server. With the NO setting, which is the default, a mapped IP (MIP) address or subnet IP (SNIP) address is used as the source IP address to initiate server side connections.

pathMonitor

Path monitoring for clustering

pathMonitorIndv

Individual Path monitoring decisions.

useproxypport

The use of client's Port.

monweight

weight of the monitor that is bound to servicegroup.

sc

Whether SureConnect is enabled on this service or not.

sp

Enable surge protection for the service group.

rtspSessionidRemap

Enable RTSP session ID mapping for the service group.

cltTimeout

Time, in seconds, after which to terminate an idle client connection.

svrTimeout

Time, in seconds, after which to terminate an idle server connection.

CKA

Enable client keep-alive for the service group.

TCPB

Enable TCP buffering for the service group.

CMP

Enable compression for the specified service.

maxBandwidth

Maximum bandwidth, in Kbps, allocated for all the services in the service group.

state

Monitor state.

svrState

The state of the service

delay

The remaining time in seconds for the service to be disabled

IP

IP Address.

serverName

The name of the server to be changed.

monitorName

Monitor name.

monThreshold

Minimum sum of weights of the monitors that are bound to this service. Used to determine whether to mark a service as UP or DOWN.

monState

The running state of the monitor on this service.

weight

weight of the monitor that is bound to servicegroup.

CustomServerID

The identifier for this IP:Port pair. Used when the persistency type is set to Custom Server ID.

serverID

The identifier for the service. This is used when the persistency type is set to Custom Server ID.

monStatCode

The code indicating the monitor response.

monStatParam1

First parameter for use with message code.

monStatParam2

Second parameter for use with message code.

monStatParam3

Third parameter for use with message code.

monitorTotalProbes

Total number of probes sent to monitor this service.

monitorTotalFailedProbes

Total number of failed probes

monitorCurrentFailedProbes

Total number of currently failed probes

downStateFlush

Flush all active transactions associated with all services in the service group whose state transitions from UP to DOWN. Do not enable this option for applications that must complete their transactions.

lastresponse

The string form of monstatcode.

stateChangeTimeSec

Time when last state change occurred. Seconds part.

stateChangeTimeMemSec

Time when last state change occurred. Milliseconds part.

timeSinceLastStateChange

Time in milliseconds since the last state change.

ticksSinceLastStateChange

Time in 10 millisecond ticks since the last state change.

StateUpdateReason

Checks state update reason on the secondary node.

CIMonOwner

Tells the mon owner of the service.

CIMonView

Tells the view id of the monitoring owner.

groupCount

Servicegroup Count

comment

Any information about the service group.

tcpProfileName

Name of the TCP profile that contains TCP configuration settings for the service group.

httpProfileName

Name of the HTTP profile that contains HTTP configuration settings for the service group.

hashId

The hash identifier for the service. This must be unique for each service. This parameter is used by hash based load balancing methods.

riseApbrStatsMsgCode

The code indicating the rise apbr status.

riseApbrStatsMsgCode2

The code indicating other rise stats.

graceFul

Indicates graceful shutdown of the service. System will wait for all outstanding connections to this service to be closed before disabling the service.

healthMonitor

Monitor the health of this service. Available settings function as follows:

YES - Send probes to check the health of the service.

NO - Do not send probes to check the health of the service. With the NO option, the appliance shows the service as UP at all times.

appflowLog

Enable logging of AppFlow information for the specified service group.

netProfile

Network profile for the service group.

autoScale

Auto scale option for a servicegroup

memberPort

member port

serviceIPstr

This field has been introduced to show the dbs services ip

serviceGroupEntName2**passive**

Indicates if load monitor is passive. A passive load monitor does not remove service from LB decision when threshold is breached.

serviceGroupeffectivestate

Indicates the effective servicegroup state based on the state of the bound service items.If all services are UP the effective state is UP, if all are DOWN its DOWN,if all are OFS its OFS.If atleast one serviceis UP and rest are either DOWN or OFS, the effective state is PARTIAL-UP.If atleast one bound service is DOWN and rest are OFS the effective state is PARTIAL DOWN.

devno**count****stateflag**

stat serviceGroup

Displays configuration statistics of the specified service group or all the service groups configured on the appliance.

Synopsys

stat serviceGroup [<serviceGroupName>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

serviceGroupName

Name of the service group for which to display settings.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

State

Current state of the server. Possible values are UP, DOWN, UNKNOWN, OFS(Out of Service), TROFS (Transition Out of Service), TROFS_DOWN(Down When going Out of Service)

Service type (Type)

The service type of this service. Possible values are ADNS, DNS, MYSQL, RTSP, SSL_DIAMETER, ADNS_TCP, DNS_TCP, NNTP, SIP_UDP, SSL_TCP, ANY, FTP, RADIUS, SNMP, TCP, DHCPRA, HTTP, RDP, SSL, TFTP, DIAMETER, MSSQL, RPCSVR, SSL_BRIDGE, UDP

rename serviceGroup

Renames a service group.

Synopsys

```
rename serviceGroup <serviceName>@ <newName>@
```

Arguments

serviceName

Existing name of the service group.

newName

New name for the service group.

Example

```
rename service svcgrp1 svcgrp-new1
```

serviceGroupMember

The following operations can be performed on "serviceGroupMember":

stat serviceGroupMember

Display statistics of a service group member.

Synopsys

```
stat serviceGroupMember <serviceGroupName> (<IP> | <serverName>) <port> [-detail] [-fullValues] [-ntimes  
<positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

serviceGroupName

Displays statistics for the specified service group. Name of the service group. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at sign (@), equal sign (=), and hyphen (-) characters.

CLI Users: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my servicegroup" or 'my servicegroup').

IP

IP address of the service group. Mutually exclusive with the server name parameter.

serverName

Name of the server. Mutually exclusive with the IP address parameter.

port

Port number of the service group member.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Average server TTFB (SvrTTFB)

Average TTFB between the NetScaler appliance and the server. TTFB is the time interval between sending the request packet to a service and receiving the first response from the service

IP address (IP)

The IP address on which the service is running.

Port (port)

The port on which the service is running.

Service type (Type)

The service type of this service. Possible values are ADNS, DNS, MYSQL, RTSP, SSL_DIAMETER, ADNS_TCP, DNS_TCP, NNTP, SIP_UDP, SSL_TCP, ANY, FTP, RADIUS, SNMP, TCP, DHCPRA, HTTP, RDP, SSL, TFTP, DIAMETER, MSSQL, RPCSVR, SSL_BRIDGE, UDP

State

Current state of the server. Possible values are UP, DOWN, UNKNOWN, OFS(Out of Service), TROFS (Transition Out of Service), TROFS_DOWN(Down When going Out of Service)

Requests (Req)

Total number of requests received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

Responses (Rsp)

Number of responses received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

Request bytes (Reqb)

Total number of request bytes received on this service or virtual server.

Response bytes (Rspb)

Number of response bytes received by this service or virtual server.

Current client connections (CIntConn)

Number of current client connections.

Requests in surge queue (SurgeQ)

Number of requests in the surge queue.

Current server connections (SvrConn)

Number of current connections to the actual servers behind the virtual server.

Current Server Est connections (SvrEstConn)

Number of server connections in ESTABLISHED state.

Connections in reuse pool (ReuseP)

Number of requests in the idle queue/reuse pool.

Maximum server connections (MaxConn)

Maximum open connections allowed on this service.

servicegroupbindings

The following operations can be performed on "servicegroupbindings":

show servicegroupbindings

Displays servicegroup information followed by vservers bound to it.

Synopsys

```
show servicegroupbindings <serviceGroupName>
```

Arguments

serviceGroupName

The name of the service.

Outputs

IPAddress

The IP address of the vserver.

port

The port of the vserver.

state

The state of the service group

svrState

The state of the vserver

vServerName

The name of the vserver.

stateflag

devno

count

svcbindings

The following operations can be performed on "svcbindings":

show svcbindings

Displays a list of all virtual servers to which the service is bound.

Synopsys

```
show svcbindings <serviceName>
```

Arguments

serviceName

The name of the service.

Outputs

IPAddress

The IP address of the vserver.

port

The port of the vserver.

svrState

The state of the vserver

vServerName

The name of the vserver.

stateflag

devno

count

uiinternal

The following operations can be performed on "uiinternal":

[set](#) | [unset](#) | [show](#)

set uiinternal

set uiinternal data for the entities

Synopsys

```
set uiinternal <entityType> <name> [-template <string>] [-comment <string>] [-rule <string>]
```

Arguments

entityType

The entity type of UI internal data

Possible values: LBVSERVER, GSLBVSERVER, CRVSERVER, VPNVSERVER, CSVSERVER, AUTHENTICATIONVSERVER, SERVER, SERVICE, SERVICEGROUP, GSLBSERVICE, EXPRESSION, VPNUURL

name

The entity name

template

The application template associated with entity

comment

The application template associated with entity

rule

rules associated with entity

Example

```
set uiinternal lbvserver v1 -template appl
```

unset uiinternal

unset uiinternal for the entities. Refer to the set uiinternal command for meanings of the arguments.

Synopsys

```
unset uiinternal <entityType> <name> [-template] [-comment] [-rule] [-all]
```

Example

```
unset uiinternal lbvserver v1 -template appl
```

show uiinternal

display all UI internal data information for the entities

Synopsys

```
show uiinternal [<entityType>] [<name>]
```

Arguments

entityType

The entity type of UI internal data

Possible values: LBVSERVER, GSLBVSERVER, CRVSERVER, VPNVSERVER, CSVSERVER, AUTHENTICATIONVSERVER, SERVER, SERVICE, SERVICEGROUP, GSLBSERVICE, EXPRESSION, VPNURL

name

The entity name

Outputs

template

The template associated with the entity

comment

The comment associated with the entity

uiinfo

The uiinfo associated with the entity

rule

The rule associated with the entity

devno

count

stateflag

Example

```
show uiinternal LBVSERVER v1
```

vserver

The following operations can be performed on "vserver":

rm | **set** | **unset** | **enable** | **disable** | **show**

rm vserver

Use this command to remove a virtual server. NOTE: This command is deprecated. This command is deprecated in 10.0, instead you can use commands such as `rm lb vserver`

Synopsys

Arguments

name

The name of the virtual server to be removed.

Example

```
rm vserver lb_vip To remove multiple vservers, use the following command: rm vserver lb_vip
```

set vserver

Use this command to modify the parameters for an existing virtual server. NOTE: This command is deprecated. This command is deprecated in 10.0, instead you can use commands such as `set lb vserver`

Synopsys

Arguments

name

The name of the virtual server for which the parameters are to be set.

pushVserver

The lb vserver of type PUSH/SSL_PUSH to which server pushes the updates received on the client facing non-push lb vserver.

Example

```
set vserver lb_vip -backupVServerName bkvip_lbvip To set backup vserver for multiple vservers
```

unset vserver

Use this command to unset the backup virtual server or the redirectURL that has been set on the virtual server..Refer to the set vserver command for meanings of the arguments.NOTE: This command is deprecated.

Synopsys

Example

```
unset vserver lb_vip -backupVServer To unset the backup vserver for multiple vservers
```

enable vserver

Use this command to enable a virtual server. Note: Virtual servers, when added, are enabled by default. NOTE: This command is deprecated. This command is deprecated in 10.0, instead you can use commands such as enable lb vserver

Synopsys

Arguments

name

The name of the virtual server to be enabled.

Example

```
enable vserver lb_vip
```

 To enable multiple vservers, use the following command: `enable`

disable vserver

Use this command to disable (take out of service) a virtual server. NOTE: This command is deprecated. This command is deprecated in 10.0, instead you can use commands such as disable lb vserver

Synopsys

Arguments

name

The name of the virtual server to be disabled.

Notes:

1. The system will continue to respond to ARP and/or ping requests for the IP address of this virtual server.
2. As the virtual server is still configured in the system, you can enable the virtual server using the `###enable vserver###` command.

Example

```
disable vserver lb_vip
```

 To disable multiple vservers, use the following command: `disable`

show vserver

Displays information about all virtual servers configured on the appliance.

Synopsys

```
show vserver
```

Example

```
show vserver lb_vip
```

Content Accelerator Commands

The entities on which you can perform NetScaler CLI operations:

- o `ca`
- o `ca action`
- o `ca global`
- o `ca policy`
- o `ca stats`

ca

The following operations can be performed on "ca":

stat ca

Shows CA performance statistics.

Synopsys

```
stat ca [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

unidentified device bytes ratio (caunidentifiedbytesHR)

This tells the hit ratio of unidentified bytes

ios bytes ratio (caiosBytesHR)

This tells the hit ratio of IOS bytes

Other device bytes ratio (caotherBytesHR)

This tells the hit ratio of Other device

Laptop/desktop bytes ratio (calaptpdsctpBytesHR)

This tells the hit ratio of laptop_desktop bytes

Android bytes ratio (caandroidBytesHR)

This tells the hit ratio of 3GP bytes

3GP bytes ratio (ca3GPcacheBytesHR)

This tells the hit ratio of 3GP bytes

MP4 bytes ratio (CaMP4cacheBytesHR)

This tells the hit ratio of MP4 bytes

FLV bytes ratio (CaFLVcacheBytesHR)

This tells the hit ratio of FLV bytes

AAC bytes ratio (CaAACcachebytesHR)

This tells the hit ratio AAC bytes served

ADTS bytes ratio (CaADTScachebytesHR)

This tells the byte hit ratio of ADTS bytes

AppleLive PL bytes ratio (CaAppleLiveStrmgPlcacheBytesHR)

This tells the byte hit ratio of AppleLive Playlist bytes

AppleLiveStream bytes ratio (CaAppleLiveStrmngcacheBytesHR)

This tells the total number of AppleLive bytes

MSFT SmoothStreamPL bytes ratio (CaMsftSmthStrmingPICaBytesHR)

This tells the byte hit ratio of MicrosoftSmoothStreaming Playlist bytes

MSFT SmoothStream bytes ratio (CaMsftSmthStrmingiCaBytesHR)

This tells the Bytes hit ratio of MicrosoftSmoothStreaming bytes .

unidentified devices (caunidentifiedHR)

This tells the hit ratio of android requests

IOS (caiosHR)

This tells the hit ratio of ios requests

Other device (caotherHR)

This tells the hit ratio of other mobile device requests

Laptop/desktop (calaptopdesktopHR)

This tells the hit ratio of laptop/desktop requests

Android (caandroidHR)

This tells the hit ratio of android requests

3GP (ca3GPHR)

This tells the total number of 3GP requests

MP4 (CacMP4HR)

This tells the hit ratio of MP4 requests

FLV (CacFLVHR)

This tells the hit ratio of FLV requests

AAC (CacAACHR)

This tells the hit ratio of AAC requests

ADTS (CacADTSHR)

This tells the Hit Ratio of ADTS requests

AppleLive PL (CacAppleLiveStreamingPlaylistiHR)

This tells the hit ratio of AppleLive Playlist requests

AppleLiveStream (CacAppleLiveStreamingHR)

This tells the hit ratio of AppleLive requests

MSFT SmoothStreamPL (CacMsftSmthStrmPIHR)

This tells the hit ratio of MicrosoftSmoothStreaming Playlist requests

MSFT SmoothStream (MsftSmthStrmHR)

This tells the hit ratio of MicrosoftSmoothStreaming requests

Total Other Objects (caOther)**Total Video Objects (caVideo)****Total Audio Objects (caAudio)****Total objects stored in cache (catotobj)****Hits being served (CaHit)**

This number should be close to the number of hits being served currently.

Total fetch request from cache (Catotgetobjres)**Total Lookup Requests (CaReq)****MSFT SmoothStream objects (MsftSmthStrm)**

This tells the total number of MicrosoftSmoothStreaming requests served by NS

MSFT SmoothStream hits (CacMstSmthStrm)

This tells the total number of MicrosoftSmoothStreaming requests served from cache

MSFT SmoothStreamPL objects (MsftSmthStreamingPI)

This tells the total number of MicrosoftSmoothStreaming Playlist requests served by NS

MSFT SmoothStreamPL hits (CacMsftSmthStrmPI)

This tells the total number of MicrosoftSmoothStreaming Playlist requests served from cache

AppleLiveStream objects (AppleLiveStrm)

This tells the total number of AppleLive requests served by NS

AppleLiveStream hits (CacAppleLiveStreaming)

This tells the total number of AppleLive requests served from cache

AppleLive PL objects (AppleLiveStrmPI)

This tells the total number of AppleLive Playlist requests served by NS

AppleLive PL hits (CacAppleLiveStreamingPlaylist)

This tells the total number of AppleLive Playlist requests served from cache

ADTS objects (CaADTS)

This tells the total number of ADTS requests served by NS

ADTS hits (CacADTS)

This tells the total number of ADTS requests served from cache

AAC objects (CaAAC)

This tells the total number of AAC requests served by NS

AAC hits (CacAAC)

This tells the total number of AAC requests served from cache

FLV objects (CaFLV)

This tells the total number of FLV requests served by NS

FLV hits (CacFLV)

This tells the total number of FLV requests served from cache

MP4 objects (CaMP4)

This tells the total number of MP4 requests served by NS

MP4 hits (CacMP4)

This tells the total number of MP4 requests served from cache

3GP Objects (ca3GP)

This tells the total number of 3GP requests served by NS

3GP hits (ca3GP)

This tells the hit ratio of 3GP requests served from cache

Total MSFT SmoothStream Bytes (CaMsftSthStrmBytes)

This tells the total number of MicrosoftSmoothStreaming bytes served by NS

CA MSFT SmoothStream Bytes (CaMicrosoftSmoothStreamingiCacheBytes)

This tells the total number of MicrosoftSmoothStreaming bytes served from cache.

Total MSFT SmoothStreamPL Bytes (CaMisftSmthStrmPIBytes)

This tells the total number of MicrosoftSmoothStreaming Playlist bytes served by NS

CA MSFT SmoothStreamPL Bytes (CaMicrosoftSmoothStreamingPlaylistCacheBytes)

This tells the total number of MicrosoftSmoothStreaming Playlist bytes served from cache

Total AppleLiveStream Bytes (CaAppleLiveStreamingBytes)

This tells the total number of AppleLive bytes served by NS

CA AppleLiveStream Bytes (CaAppleLiveStreamingcacheBytes)

This tells the total number of AppleLive bytes served from cache

Total AppleLive PL Bytes (CaAppleLiveStreamingPlaylistBytes)

This tells the total number of AppleLive Playlist bytes served by NS

CA AppleLivePL Bytes (CaAppleLiveStreamingPlaylistcacheBytes)

This tells the total number of AppleLive Playlist bytes served from cache

Total ADTS bytes (CaADTSbytes)

This tells the total number of ADTS bytes served by NS

CA ADTS bytes (CaADTScachebytes)

This tells the total number of ADTS bytes served from cache

Total AAC bytes (CaAACbytes)

This tells the total number of AAC bytes served by NS

CA AAC bytes (CaAACcachebytes)

This tells the total number of AAC bytes served from cache

Total FLV bytes (CaFLVBytes)

This tells the total number of FLV bytes served by NS

CA FLV bytes (CaFLVcacheBytes)

This tells the total number of FLV bytes served from cache

Total MP4 bytes (CaMP4Bytes)

This tells the total number of MP4 bytes served by NS

CA MP4 bytes (CaMP4cacheBytes)

This tells the total number of MP4 bytes served from cache

Total 3GP Bytes (ca3GPBytes)

This tells the total number of 3GP bytes served by NS

CA 3GP Bytes (ca3GPcacheBytes)

This tells the total number of 3GP bytes served from cache

Android requests (caandroid)

Total number of android requests to netscaler

Laptop/Desktop requests (calaptopDesktop)

Total number of laptop/desktop requests to netscaler

IOS requests (caios)

Total number of iOS requests to netscaler

Other requests (caother)

Total number of other mobile device requests to netscaler

Unidentified requests (caunidentified)

Total number of unidentified requests to netscaler

Android hits (caandroidcache)

This tells android requests served from cache

IOS hits (caioscache)

This tells iOS requests served from cache

Other device hits (caothercache)

This tells Other device requests served from cache

laptop/desktop hits (calaptopdesktopcache)

This tells laptop/desktop requests served from cache

Unidentified device hits (caunidentifiedcache)

This tells unidentified device requests served from cache

Total Android Bytes (caandroidBytes)

This tells the total number of Android bytes served by NS

Total IOS Bytes (caiosBytes)

This tells the total number of IOS bytes served by NS

Total Other device Bytes (caotherBytes)

This tells the total number of Other mobile device bytes served by NS

Total Laptop/desktop Bytes (calaptopdesktopBytes)

This tells the total number of Laptop/desktop bytes served by NS

Total unidentified device Bytes (caunidentifiedBytes)

This tells the total number of unidentified device bytes served by NS

CA Android Bytes (caandroidcacheBytes)

This tells the total number of Android bytes served from cache

CA IOS Bytes (caioscacheBytes)

This tells the total number of IOS bytes served from cache

CA Other device Bytes (caothercacheBytes)

This tells the total number of other device bytes served from cache

CA Laptop/desktop Bytes (calaptpdesktopBytes)

This tells the total number of Laptop/desktop bytes served from cache

CA unidentified device Bytes (caunidentifiedBytes)

This tells the total number of unidentified device bytes served from cache

ca action

The following operations can be performed on "ca action":

add | **show** | **set** | **unset** | **rm** | **rename**

add ca action

Creates a content adaptation action. This action must later be invoked in the 'add ca policy' command.

Synopsis

add ca action <name> [-accumResSize <KBytes>] [-lbvserver <string>] [-comment <string>] -type <type>

Arguments

name

Name of the content adaptation action. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

accumResSize

Size of the data, in KB, that the server must respond with. The NetScaler uses this data to compute a hash which is then used to lookup within the T2100 appliance.

lbvserver

Name of the load balancing virtual server that has the T2100 appliances as services.

comment

Information about the content adaptation action.

type

Specifies whether the NetScaler must lookup for the response on the T2100 appliance or serve the response directly from the server.

Possible values: nlookup, lookup, noop

show ca action

Displays information about a content adaptation action. If no name is specified, this command displays information of all available content adaptation actions.

Synopsis

show ca action [<name>]

Arguments

name

Name of the content accelerator action.

Outputs

type

Type of content accelerator action.

stateflag

accumResSize

Size of the data, in KB, that the server must respond with. The NetScaler uses this data to compute a hash which is then used to lookup within the T2100 appliance.

lbvserver

Name of the load balancing virtual server that has the T2100 appliances as services.

isDefault

A value of true is returned if it is a default content accelerator action.

hits

The number of times the action has been taken.

builtin

Flag to determine whether content accelerator action is built-in or not

flags**comment**

Information about the content adaptation action.

devno**count**

Example

```
1. show ca action 2. show ca action act_insert
```

set ca action

Modifies the specified parameters of a Content Accelerator action.

Synopsys

```
set ca action <name> [-accumResSize <KBytes>] [-type <type>] [-lbvserver <string>] [-comment <string>]
```

Arguments

name

Name of the Content Accelerator policy to modify.

accumResSize

Size of the data, in KB, that the server must respond with. The NetScaler uses this data to compute a hash which is then used to lookup within the T2100 appliance.

type

Specifies whether the NetScaler must lookup for the response on the T2100 appliance or serve the response directly from the server.

Possible values: nlookup, lookup, noop

lbvserver

Name of the load balancing virtual server that has the T2100 appliances as services.

comment

Information about the content adaptation action.

Example

```
set ca action caactl -accumresize 43"
```

unset ca action

Use this command to remove ca action settings. Refer to the set ca action command for meanings of the arguments.

Synopsys

```
unset ca action <name> [-accumResSize] [-type] [-lbvserver] [-comment]
```

rm ca action

Removes a ca action.

Synopsys

```
rm ca action <name>
```

Arguments

name

Name of the Content Accelerator action to remove.

Example

```
rm ca action act_before
```

rename ca action

Renames a Content Accelerator action.

Synopsys

```
rename ca action <name>@ <newName>@
```

Arguments

name

Existing name of the Content Accelerator action.

newName

New name for the ContentAdaptation action.

Must begin with a letter, number, or the underscore character (), and must contain only letters, numbers, and the hyphen (-), period (.) hash (#), space (), at (@), equals (=), colon (:), and underscore characters. Can be changed after the ContentAdaptation policy is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my ContentAdaptation action" or ?my ContentAdaptation action?).!.

Example

```
rename ca action oldname newname
```

ca global

The following operations can be performed on "ca global":

[bind](#) | [unbind](#) | [show](#)

bind ca global

Activates the specified content accelerator policy for all requests sent to the NetScaler appliance.

Synopsys

```
bind ca global -policyName <string> -priority <positive_integer> [-gotoPriorityExpression <expression>] [-type <type>]
```

Arguments

policyName

Name of the content accelerator policy.

priority

Specifies the priority of the content accelerator policy.

Minimum value: 0

gotoPriorityExpression

type

Example

```
i) bind ca global pol9 9
```

unbind ca global

Unbind the specified content accelerator policy from ContentAccelerator global.

Synopsys

```
unbind ca global <policyName> [-type <type>] [-priority <positive_integer>]
```

Arguments

policyName

Name of the policy to unbind.

type

The bindpoint from which the policy is to be unbound.

Possible values: REQ_OVERRIDE, REQ_DEFAULT, RES_OVERRIDE, RES_DEFAULT

priority

Priority of the NOPOLICY to be unbound.

Minimum value: 1

Maximum value: 2147483647

Example


```
unbind ca global pol9
```

show ca global

Shows the content adaptation policies that are globally-bound to the NetScaler appliance.

Synopsys

```
show ca global [-type <type>]
```

Arguments

type

Outputs

stateflag

policyName

Name of the content accelerator policy.

priority

Specifies the priority of the content accelerator policy.

gotoPriorityExpression

flowType

flowtype of the bound content accelerator policy.

numpol

Number of polices bound to label.

flags

devno

count

Example

```
show ca global
```

ca policy

The following operations can be performed on "ca policy":

add | **show** | **rm** | **set** | **unset** | **rename**

add ca policy

Creates a content adaptation policy. This policy must later be invoked globally or at a content switching or load balancing virtual server.

Synopsys

```
add ca policy <name> -rule <expression> -action <string> [-undefAction <string>] [-comment <string>] [-logAction <string>]
```

Arguments

name

Name for the content adaptation policy. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Can be changed after the policy is created.

rule

Expression that determines which requests or responses match the content adaptation policy. When specifying the rule in the CLI, the description must be enclosed within double quotes.

action

Name of content adaptation action to be executed when the rule is evaluated to true.

undefAction

comment

Information about the content adaptation policy.

logAction

Name of messagelog action to use when a request matches this policy.

show ca policy

Displays information about a content adaptation policy. If no name is specified, this command displays information of all available content adaptation policies.

Synopsys

```
show ca policy [<name>]
```

Arguments

name

Name of the content adaptation policy to be displayed.

Outputs

stateflag

undefAction

Undef Action associated with the policy.

activePolicy

Indicates whether policy is bound or not.

boundTo

Location where policy is bound

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

vserverType**action**

Content adaptation action associated with the policy.

rule

Rule of the policy.

hits

Number of hits.

undefHits

Number of Undef hits.

isDefault

A value of true is returned if it is a default ContentAdaptationpolicy.

priority

Specifies the priority of the policy.

bindPolicyType**comment**

Information about the content adaptation policy.

logAction

Name of messagelog action to use when a request matches this policy.

devno**count**

Example

```
show ca policy
```

rm ca policy

Removes a content adaptation policy.

Synopsys

```
rm ca policy <name>
```

Arguments

name

Name of the content adaptation policy to be removed.

Example

```
rm ca policy pol9
```

set ca policy

Modifies the parameters of a content adaptation policy.

Synopsys

```
set ca policy <name> [-rule <expression>] [-action <string>] [-comment <string>] [-logAction <string>] [-undefAction <string>]
```

Arguments

name

Name of the content accelerator policy to be modified.

rule

Expression that determines which requests or responses match the content adaptation policy. When specifying the rule in the CLI, the description must be enclosed within double quotes.

action

Name of content adaptation action to be executed when the rule is evaluated to true.

comment

Information about the content adaptation policy.

logAction

Name of messagelog action to use when a request matches this policy.

undefAction

Example

```
set ca policy pol9 -rule "HTTP.REQ.HEADER(\\\\\\"header\\\\\\").CONTAINS(\\\\\\"qh2\\\\\\")"
```

unset ca policy

Removes the settings of an existing content accelerator policy. Attributes for which a default value is available revert to their default values. See the set content accelerator policy command for a description of the parameters..Refer to the set ca policy command for meanings of the arguments.

Synopsys

```
unset ca policy <name> [-comment] [-logAction] [-undefAction]
```

Example

```
unset ca policy pol9 -undefAction
```

rename ca policy

Renames content accelerator policy.

Synopsys

```
rename ca policy <name>@ <newName>@
```

Arguments

name

Existing name of the content accelerator policy.

newName

New name for the content accelerator policy

Example

```
rename ca policy oldname newname
```

ca stats

The following operations can be performed on "ca stats":

show ca stats

show ca stats is an alias for stat ca

Synopsys

show ca stats - alias for 'stat ca'

Cache Commands

The entities on which you can perform NetScaler CLI operations:

- o cache
- o cache contentGroup
- o cache forwardProxy
- o cache global
- o cache object
- o cache parameter
- o cache policy
- o cache policylabel
- o cache selector
- o cache stats

cache

The following operations can be performed on "cache":

stat cache

Shows Integrated Cache performance statistics.

Synopsys

```
stat cache [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

Maximum memory(KB) Deprecated (MaxMem)

Largest amount of memory the NetScaler can dedicate to caching, up to 50% of available memory. A 0 value disables caching, but the caching module continues to run.

Recent successful reval ratio(%) (RPSucRev)

Recently recorded percentage of times stored content was successfully revalidated by a 304 response rather than by a full response

Recent storable miss ratio(%) (RPctStMis)

Recently recorded ratio of store-able misses to all misses expressed as percentage.

Recent parameterized 304 hit ratio(%) (RPPHit)

Recently recorded ratio of parameterized 304 hits to all parameterized hits expressed as a percentage

Recent origin bandwidth saved(%) (RPOrBan)

Bytes served from cache divided by total bytes served to client. This ratio can be greater than 1 because of the assumption that all compression has been done in the NetScaler.

Recent hit ratio(%) (RPctHit)

Recently recorded cache hit ratio expressed as percentage

Recent byte hit ratio(%) (RPcByHit)

Recently recorded cache byte hit ratio expressed as percentage. Here we define byte hit ratio as ((number of bytes served from the cache)/(total number of bytes served to the client)). This is the standard definition of Byte Hit Ratio. If compression is turned ON in NS then this ratio doesn't mean much. This might under or over estimate the origin-to-cache bandwidth saving (depending upon whether bytes served by CMP in NetScaler are more or less than compressed bytes served from the cache). If CMP is turned OFF in NS then this ratio is same as cacheRecentPercentOriginBandwidthSaved.

Recent 304 hit ratio(%) (RPct304Hit)

Recently recorded ratio of 304 hits to all hits expressed as percentage

Utilized memory(KB) (UtiMem)

Amount of memory the integrated cache is currently using.

Maximum memory active value(KB) (MaxMemActive)

Currently active value of maximum memory

Maximum memory(KB) (Max64Mem)

Largest amount of memory the NetScaler can dedicate to caching, up to 50% of available memory. A 0 value disables caching, but the caching module continues to run.

Poll every time hit ratio(%) (PPeHit)

Percentage of cache hits in content groups that have Poll Every Time enabled, relative to all searches of content groups with Poll Every Time enabled.

Poll every time hits (PeHit)

Number of times a cache hit was found during a search of a content group that has Poll Every Time enabled.

Parameterized 304 hit ratio(%) (PP304Hit)

Percentage of parameterized 304 hits relative to all parameterized hits.

Total parameterized hits (PHit)

Parameterized requests resulting in either a 304 or non-304 hit.

Successful reval ratio(%) (PSucRev)

Percentage of times stored content was successfully revalidated by a 304 (Object Not Modified) response rather than by a full response

Storable miss ratio(%) (PStrMiss)

Responses that were fetched from the origin, stored in the cache, and then served to the client, as a percentage of all cache misses.

Conversions to conditional req (FuToCon)

Number of user-agent requests for a cached Poll Every Time (PET) response that were sent to the origin server as conditional requests.

Successful revalidations (TSucRev)

Total number of times stored content was successfully revalidated by a 304 Not Modified response from the origin.

Revalidations (Reval)

Responses that an intervening cache revalidated with the integrated cache before serving, as determined by a Cache-Control: Max-Age header configurable in the integrated cache

Non-storable misses (NStrMiss)

Cache misses for which the fetched response is not stored in the cache. These responses match policies with a NOCACHE action or are affected by Poll Every Time.

Storable misses (StrMiss)

Cache misses for which the fetched response is stored in the cache before serving it to the client. Storable misses conform to a built-in or user-defined caching policy that contains a CACHE action.

Compressed bytes from cache (CmpBySer)

Number of compressed bytes served from the cache

Byte hit ratio(%) (PByHit)

Bytes served from the cache divided by total bytes served to the client. If compression is On in the NetScaler, this ratio may not reflect the bytes served by the compression module. If the compression is Off, this ratio is the same as cachePercentOriginBandwidthSaved.

Bytes served by cache (BySer)

Total number of bytes served from the integrated cache

Bytes served by NetScaler (RespBy)

Total number of HTTP response bytes served by NetScaler from both the origin and the cache

304 hit ratio(%) (Pct304Hit)

304 responses as a percentage of all responses that the NetScaler served.

Marker objects (NumMark)

Marker objects created when a response exceeds the maximum or minimum size for entries in its content group or has not yet received the minimum number of hits required for items in its content group.

Origin bandwidth saved(%) (POrBan)

Percentage of origin bandwidth saved, expressed as number of bytes served from the integrated cache divided by all bytes served. The assumption is that all compression is done in the NetScaler.

Hit ratio(%) (PctHit)

Cache hits as percentage of the total number of requests

Misses (TotMiss)

Intercepted HTTP requests requiring fetches from origin server.

Hits (TotHit)

Responses served from the integrated cache. These responses match a policy with a CACHE action.

Requests (CacReq)

Total cache hits plus total cache misses.

Cached objects (NumCac)

Responses currently in integrated cache. Includes responses fully downloaded, in the process of being downloaded, and expired or flushed but not yet removed.

Hits being served (CacHit)

This number should be close to the number of hits being served currently.

Misses being handled (CurMiss)

Responses fetched from the origin and served from the cache. Should approximate storable misses. Does not include non-storable misses.

Non-304 hits (Non304Hit)

Total number of full (non-304) responses served from the cache. A 304 status code indicates that a response has not been modified since the last time it was served

304 hits (304Hit)

Object not modified responses served from the cache. (Status code 304 served instead of the full response.)

sql hits (sqlHit)

sql response served from cache

Expire at last byte (ExpLa)

Instances of content expiring immediately after receiving the last body byte due to the Expire at Last Byte setting for the content group.

Flashcache misses (FIMi)

Number of requests to a content group with flash cache enabled that were cache misses. Flash cache distributes the response to all the clients in a queue.

Flashcache hits (FIHi)

Number of requests to a content group with flash cache enabled that were cache hits. The flash cache setting queues requests that arrive simultaneously and distributes the response to all the clients in the queue.

Parameterized inval requests (PInReq)

Requests matching a policy with an invalidation (INVALID) action and a content group that uses an invalidation selector or parameters.

Full inval requests (NPInReq)

Requests that match an invalidation policy where the invalGroups parameter is configured and expires one or more content groups.

Inval requests (INStrMis)

Requests that match an invalidation policy and result in expiration of specific cached responses or entire content groups.

Parameterized requests (PReq)

Total number of requests where the content group has hit and invalidation parameters or selectors.

Parameterized non-304 hits (PN304Hit)

Parameterized requests resulting in a full response (not status code 304: Object Not Updated) served from the cache.

Parameterized 304 hits (P304Hit)

Parameterized requests resulting in an object not modified (status code 304) response.

Poll every time requests (PeReq)

Requests that triggered a search of a content group that has Poll Every Time (PET) enabled (always consult the origin server before serving cached data).

Memory allocation failures (ErrMem)

Total number of times the cache failed to allocate memory to store responses.

Largest response so far(B) (LarResp)

Size, in bytes, of largest response sent to client from the cache or the origin server.

Compressed bytes transmitted

Number of bytes the NetScaler sends to the client after compressing the response from the server.

Compressible bytes received

Number of bytes that can be compressed, which the NetScaler receives from the server. This gives the content length of the response that the NetScaler receives from server.

Response bytes received (HTRspbRx)

Total number of bytes of HTTP response data received.

cache contentGroup

The following operations can be performed on "cache contentGroup":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#) | [expire](#) | [flush](#) | [stat](#) | [save](#)

add cache contentGroup

Creates a new content group for grouping cached objects on the basis of some unique property.

Synopsys

```
add cache contentGroup <name> [-weakPosRelExpiry <secs> | -relExpiry <secs> | -relExpiryMilliSec <msecs> | -absExpiry <HH:MM> ... | -absExpiryGMT <HH:MM> ...] [-heurExpiryParam <positive_integer>] [-weakNegRelExpiry <secs>] [(-hitParams <string> ... [-ignoreParamValueCase ( YES | NO ) | -hitSelector <string> | -invalSelector <string>] [-matchCookies ( YES | NO )])] [-invalParams <string> ... [-invalRestrictedToHost ( YES | NO )]] [-pollEveryTime ( YES | NO )] [-ignoreReloadReq ( YES | NO )] [-removeCookies ( YES | NO )] [-prefetch ( YES | NO )] [-prefetchPeriod <secs> | -prefetchPeriodMilliSec <msecs>]] [-prefetchMaxPending <positive_integer>] [-flashCache ( YES | NO )] [-expireAtLastByte ( YES | NO )] [-insertVia ( YES | NO )] [-insertAge ( YES | NO )] [-insertETag ( YES | NO )] [-cacheControl <string>] [-quickAbortSize <KBytes>] [-minResSize <KBytes>] [-maxResSize <KBytes>] [-memLimit <MBytes>] [-ignoreReqCachingHdrs ( YES | NO )] [-minHits <integer>] [-alwaysEvalPolicies ( YES | NO )] [-persistHA ( YES | NO )] [-pinned ( YES | NO )] [-lazyDnsResolve ( YES | NO )] [-type <type>]
```

Arguments

name

Name for the content group. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the content group is created.

weakPosRelExpiry

Relative expiry time, in seconds, for expiring positive responses with response codes between 200 and 399. Cannot be used in combination with other Expiry attributes. Similar to -relExpiry but has lower precedence.

Default value: -1

Maximum value: 31536000

heurExpiryParam

Heuristic expiry time, in percent of the duration, since the object was last modified.

Default value: -1

Minimum value: 0

Maximum value: 100

relExpiry

Relative expiry time, in seconds, after which to expire an object cached in this content group.

Default value: -1

Maximum value: 31536000

relExpiryMilliSec

Relative expiry time, in milliseconds, after which to expire an object cached in this content group.

Default value: -1

Maximum value: 86400000

absExpiry

Local time, up to 4 times a day, at which all objects in the content group must expire.

CLI Users:

For example, to specify that the objects in the content group should expire by 11:00 PM, type the following command: `add cache contentgroup <contentgroup name> -absexpiry 23:00`

To specify that the objects in the content group should expire at 10:00 AM, 3 PM, 6 PM, and 11:00 PM, type: `add cache contentgroup <contentgroup name> -absexpiry 10:00 15:00 18:00 23:00`

absExpiryGMT

Coordinated Universal Time (GMT), up to 4 times a day, when all objects in the content group must expire.

weakNegRelExpiry

Relative expiry time, in seconds, for expiring negative responses. This value is used only if the expiry time cannot be determined from any other source. It is applicable only to the following status codes: 307, 403, 404, and 410.

Default value: -1

Maximum value: 31536000

hitParams

Parameters to use for parameterized hit evaluation of an object. Up to 128 parameters can be specified. Mutually exclusive with the Hit Selector parameter.

invalParams

Parameters for parameterized invalidation of an object. You can specify up to 8 parameters. Mutually exclusive with `invalSelector`.

ignoreParamValueCase

Ignore case when comparing parameter values during parameterized hit evaluation. (Parameter value case is ignored by default during parameterized invalidation.)

Possible values: YES, NO

Default value: VAL_NOT_SET

matchCookies

Evaluate for parameters in the cookie header also.

Possible values: YES, NO

Default value: VAL_NOT_SET

invalRestrictedToHost

Take the host header into account during parameterized invalidation.

Possible values: YES, NO

Default value: VAL_NOT_SET

pollEveryTime

Always poll for the objects in this content group. That is, retrieve the objects from the origin server whenever they are requested.

Possible values: YES, NO

Default value: NO

ignoreReloadReq

Ignore any request to reload a cached object from the origin server.

To guard against Denial of Service attacks, set this parameter to YES. For RFC-compliant behavior, set it to NO.

Possible values: YES, NO

Default value: YES

removeCookies

Remove cookies from responses.

Possible values: YES, NO

Default value: YES

prefetch

Attempt to refresh objects that are about to go stale.

Possible values: YES, NO

Default value: YES

prefetchPeriod

Time period, in seconds before an object's calculated expiry time, during which to attempt prefetch.

Default value: -1

Maximum value: 4294967294

prefetchPeriodMilliSec

Time period, in milliseconds before an object's calculated expiry time, during which to attempt prefetch.

Default value: -1

Maximum value: 4294967290

prefetchMaxPending

Maximum number of outstanding prefetches that can be queued for the content group.

Default value: -1

Minimum value: 0

Maximum value: 4294967294

flashCache

Perform flash cache. Mutually exclusive with Poll Every Time (PET) on the same content group.

Possible values: YES, NO

Default value: NO

expireAtLastByte

Force expiration of the content immediately after the response is downloaded (upon receipt of the last byte of the response body). Applicable only to positive responses.

Possible values: YES, NO

Default value: NO

insertVia

Insert a Via header into the response.

Possible values: YES, NO

Default value: YES

insertAge

Insert an Age header into the response. An Age header contains information about the age of the object, in seconds, as calculated by the integrated cache.

Possible values: YES, NO

Default value: YES

insertETag

Insert an ETag header in the response. With ETag header insertion, the integrated cache does not serve full responses on repeat requests.

Possible values: YES, NO

Default value: YES

cacheControl

Insert a Cache-Control header into the response.

quickAbortSize

If the size of an object that is being downloaded is less than or equal to the quick abort value, and a client aborts during the download, the cache stops downloading the response. If the object is larger than the quick abort size, the cache continues to download the response.

Default value: 4194303

Maximum value: 4194303

minResSize

Minimum size of a response that can be cached in this content group.

Default minimum response size is 0.

Maximum value: 2097151

maxResSize

Maximum size of a response that can be cached in this content group.

Default value: 80

Maximum value: 2097151

memLimit

Maximum amount of memory that the cache can use. The effective limit is based on the available memory of the NetScaler appliance.

Default value: 65536

ignoreReqCachingHdrs

Ignore Cache-Control and Pragma headers in the incoming request.

Possible values: YES, NO

Default value: YES

minHits

Number of hits that qualifies a response for storage in this content group.

Default value: 0

alwaysEvalPolicies

Force policy evaluation for each response arriving from the origin server. Cannot be set to YES if the Prefetch parameter is also set to YES.

Possible values: YES, NO

Default value: NO

persistHA

Setting persistHA to YES causes IC to save objects in contentgroup to Secondary node in HA deployment.

Possible values: YES, NO

Default value: NO

pinned

Do not flush objects from this content group under memory pressure.

Possible values: YES, NO

Default value: NO

lazyDnsResolve

Perform DNS resolution for responses only if the destination IP address in the request does not match the destination IP address of the cached response.

Possible values: YES, NO

Default value: YES

hitSelector

Selector for evaluating whether an object gets stored in a particular content group. A selector is an abstraction for a collection of PIXL expressions.

invalSelector

Selector for invalidating objects in the content group. A selector is an abstraction for a collection of PIXL expressions.

type

The type of the content group.

Possible values: HTTP, MYSQL, MSSQL

Default value: HTTP

rm cache contentGroup

Removes the specified content group. Before removing, make sure that no cache policy has its storeInGroup attribute set to this group, otherwise the group cannot be removed.

Synopsys

rm cache contentGroup <name>

Arguments

name

Name of the content group to be removed.

set cache contentGroup

Modifies the specified attributes of the content group.

Synopsys

```
set cache contentGroup <name> [-weakPosRelExpiry <secs> | -relExpiry <secs> | -relExpiryMilliSec <msecs> | -absExpiry <HH:MM> ... | -absExpiryGMT <HH:MM> ...] [-heurExpiryParam <positive_integer>] [-weakNegRelExpiry <secs>] [-hitParams <string> ... | -hitSelector <string> | -invalSelector <string>] [-invalParams <string> ...] [-ignoreParamValueCase ( YES | NO )] [-matchCookies ( YES | NO )] [-invalRestrictedToHost ( YES | NO )] [-pollEveryTime ( YES | NO )] [-ignoreReloadReq ( YES | NO )] [-removeCookies ( YES | NO )] [-prefetch ( YES | NO )] [-prefetchPeriod <secs> | -prefetchPeriodMilliSec <msecs>] [-prefetchMaxPending <positive_integer>] [-flashCache ( YES | NO )] [-expireAtLastByte ( YES | NO )] [-insertVia ( YES | NO )] [-insertAge ( YES | NO )] [-insertETag ( YES | NO )] [-cacheControl <string>] [-quickAbortSize <KBytes>] [-minResSize <KBytes>] [-maxResSize <KBytes>] [-memLimit <MBytes>] [-ignoreReqCachingHdrs ( YES | NO )] [-minHits <integer>] [-alwaysEvalPolicies ( YES | NO )] [-persistHA ( YES | NO )] [-pinned ( YES | NO )] [-lazyDnsResolve ( YES | NO )]
```

Arguments

name

Name of the content group to be modified.

weakPosRelExpiry

Relative expiry time, in seconds, for expiring positive responses with response codes between 200 and 399. Cannot be used in combination with other Expiry attributes. Similar to -relExpiry but has lower precedence.

Maximum value: 31536000

heurExpiryParam

Heuristic expiry time, in percent of the duration, since the object was last modified.

Minimum value: 0

Maximum value: 100

relExpiry

Relative expiry time, in seconds, after which to expire an object cached in this content group.

Default value: -1

Maximum value: 31536000

relExpiryMilliSec

Relative expiry time, in milliseconds, after which to expire an object cached in this content group.

Default value: -1

Maximum value: 86400000

absExpiry

Local time, up to 4 times a day, at which all objects in the content group must expire.

CLI Users:

For example, to specify that the objects in the content group should expire by 11:00 PM, type the following command: add cache contentgroup <contentgroup name> -absexpiry 23:00

To specify that the objects in the content group should expire at 10:00 AM, 3 PM, 6 PM, and 11:00 PM, type: add cache contentgroup <contentgroup name> -absexpiry 10:00 15:00 18:00 23:00

absExpiryGMT

Coordinated Universal Time (GMT), up to 4 times a day, when all objects in the content group must expire.

weakNegRelExpiry

Relative expiry time, in seconds, for expiring negative responses. This value is used only if the expiry time cannot be determined from any other source. It is applicable only to the following status codes: 307, 403, 404, and 410.

Maximum value: 31536000

hitParams

Parameters to use for parameterized hit evaluation of an object. Up to 128 parameters can be specified. Mutually exclusive with the Hit Selector parameter.

invalidParams

Parameters for parameterized invalidation of an object. You can specify up to 8 parameters. Mutually exclusive with invalidSelector.

ignoreParamValueCase

Ignore case when comparing parameter values during parameterized hit evaluation. (Parameter value case is ignored by default during parameterized invalidation.)

Possible values: YES, NO

matchCookies

Evaluate for parameters in the cookie header also.

Possible values: YES, NO

invalidRestrictedToHost

Take the host header into account during parameterized invalidation.

Possible values: YES, NO

pollEveryTime

Always poll for the objects in this content group. That is, retrieve the objects from the origin server whenever they are requested.

Possible values: YES, NO

Default value: NO

ignoreReloadReq

Ignore any request to reload a cached object from the origin server.

To guard against Denial of Service attacks, set this parameter to YES. For RFC-compliant behavior, set it to NO.

Possible values: YES, NO

Default value: YES

removeCookies

Remove cookies from responses.

Possible values: YES, NO

Default value: YES

prefetch

Attempt to refresh objects that are about to go stale.

Possible values: YES, NO

Default value: YES

prefetchPeriod

Time period, in seconds before an object's calculated expiry time, during which to attempt prefetch.

Default value: -1

Maximum value: 4294967294

prefetchPeriodMilliSec

Time period, in milliseconds before an object's calculated expiry time, during which to attempt prefetch.

Default value: -1

Maximum value: 4294967290

prefetchMaxPending

Maximum number of outstanding prefetches that can be queued for the content group.

Minimum value: 0

Maximum value: 4294967294

flashCache

Perform flash cache. Mutually exclusive with Poll Every Time (PET) on the same content group.

Possible values: YES, NO

Default value: NO

expireAtLastByte

Force expiration of the content immediately after the response is downloaded (upon receipt of the last byte of the response body). Applicable only to positive responses.

Possible values: YES, NO

Default value: NO

insertVia

Insert a Via header into the response.

Possible values: YES, NO

Default value: YES

insertAge

Insert an Age header into the response. An Age header contains information about the age of the object, in seconds, as calculated by the integrated cache.

Possible values: YES, NO

Default value: YES

insertETag

Insert an ETag header in the response. With ETag header insertion, the integrated cache does not serve full responses on repeat requests.

Possible values: YES, NO

Default value: YES

cacheControl

Insert a Cache-Control header into the response.

quickAbortSize

If the size of an object that is being downloaded is less than or equal to the quick abort value, and a client aborts during the download, the cache stops downloading the response. If the object is larger than the quick abort size, the cache continues to download the response.

Maximum value: 4194303

minResSize

Minimum size of a response that can be cached in this content group.

Default minimum response size is 0.

Maximum value: 2097151

maxResSize

Maximum size of a response that can be cached in this content group.

Default value: 80

Maximum value: 2097151

memLimit

Maximum amount of memory that the cache can use. The effective limit is based on the available memory of the NetScaler appliance.

Default value: 65536

ignoreReqCachingHdrs

Ignore Cache-Control and Pragma headers in the incoming request.

Possible values: YES, NO

Default value: YES

minHits

Number of hits that qualifies a response for storage in this content group.

alwaysEvalPolicies

Force policy evaluation for each response arriving from the origin server. Cannot be set to YES if the Prefetch parameter is also set to YES.

Possible values: YES, NO

Default value: NO

persistHA

The option for IC objects to save objects to Secondary in a HA deployment. Set YES for IC to take this state.

Possible values: YES, NO

Default value: NO

pinned

The option for IC from flushing objects from this contentgroup under memory pressure. Set YES for IC to take this state.

Possible values: YES, NO

Default value: NO

lazyDnsResolve

Perform DNS resolution for responses only if the destination IP address in the request does not match the destination IP address of the cached response.

Possible values: YES, NO

Default value: YES

hitSelector

Selector for evaluating whether an object gets stored in a particular content group. A selector is an abstraction for a collection of PIXL expressions.

invalSelector

Selector for invalidating objects in the content group. A selector is an abstraction for a collection of PIXL expressions.

unset cache contentGroup

Use this command to remove cache contentGroup settings. Refer to the set cache contentGroup command for meanings of the arguments.

Synopsys

```
unset cache contentGroup <name> [-weakPosRelExpiry] [-heurExpiryParam] [-relExpiry] [-relExpiryMilliSec] [-absExpiry] [-absExpiryGMT] [-weakNegRelExpiry] [-hitParams] [-invalParams] [-ignoreParamValueCase] [-matchCookies] [-invalRestrictedToHost] [-pollEveryTime] [-ignoreReloadReq] [-removeCookies] [-prefetch] [-prefetchPeriod] [-prefetchPeriodMilliSec] [-prefetchMaxPending] [-flashCache] [-expireAtLastByte] [-insertVia] [-insertAge] [-insertETag] [-cacheControl] [-quickAbortSize] [-minResSize] [-maxResSize] [-memLimit] [-ignoreReqCachingHdrs] [-minHits] [-alwaysEvalPolicies] [-persistHA] [-pinned] [-lazyDnsResolve] [-hitSelector] [-invalSelector]
```

show cache contentGroup

Displays information about all content groups, or about the specified content group.

Synopsys

```
show cache contentGroup [<name>]
```

Arguments

name

Name of the content group about which to display information.

Outputs

flags

Flags.

type

The type of the content group.

relExpiry

The relative expiry time in seconds.

relExpiryMilliSec

Relative expiry time, in milliseconds, after which to expire an object cached in this content group.

absExpiry

Local time, up to 4 times a day, at which all objects in the content group must expire.

CLI Users:

For example, to specify that the objects in the content group should expire by 11:00 PM, type the following command: `add cache contentgroup <contentgroup name> -absexpiry 23:00`

To specify that the objects in the content group should expire at 10:00 AM, 3 PM, 6 PM, and 11:00 PM, type: `add cache contentgroup <contentgroup name> -absexpiry 10:00 15:00 18:00 23:00`

absExpiryGMT

Coordinated Universal Time (GMT), up to 4 times a day, when all objects in the content group must expire.

heurExpiryParam

Heuristic expiry time, in percent of the duration, since the object was last modified.

weakPosRelExpiry

Relative expiry time, in seconds, for expiring positive responses with response codes between 200 and 399. Cannot be used in combination with other Expiry attributes. Similar to `-relExpiry` but has lower precedence.

weakNegRelExpiry

Relative expiry time, in seconds, for expiring negative responses. This value is used only if the expiry time cannot be determined from any other source. It is applicable only to the following status codes: 307, 403, 404, and 410.

hitParams

Parameters to use for parameterized hit evaluation of an object. Up to 128 parameters can be specified. Mutually exclusive with the Hit Selector parameter.

invalParams

Parameters for parameterized invalidation of an object. You can specify up to 8 parameters. Mutually exclusive with `invalSelector`.

ignoreParamValueCase

Ignore case when comparing parameter values during parameterized hit evaluation. (Parameter value case is ignored by default during parameterized invalidation.)

matchCookies

Evaluate for parameters in the cookie header also.

invalRestrictedToHost

Take the host header into account during parameterized invalidation.

pollEveryTime

Always poll for the objects in this content group. That is, retrieve the objects from the origin server whenever they are requested.

ignoreReloadReq

Ignore any request to reload a cached object from the origin server.

To guard against Denial of Service attacks, set this parameter to YES. For RFC-compliant behavior, set it to NO.

removeCookies

Remove cookies from responses.

prefetch

Attempt to refresh objects that are about to go stale.

prefetchPeriod

Time period, in seconds before an object's calculated expiry time, during which to attempt prefetch.

prefetchPeriodMilliSec

Time period, in milliseconds before an object's calculated expiry time, during which to attempt prefetch.

prefetchCur

Current outstanding prefetches.

prefetchMaxPending

Maximum number of outstanding prefetches that can be queued for the content group.

flashCache

Perform flash cache. Mutually exclusive with Poll Every Time (PET) on the same content group.

expireAtLastByte

Force expiration of the content immediately after the response is downloaded (upon receipt of the last byte of the response body). Applicable only to positive responses.

insertVia

Insert a Via header into the response.

insertAge

Insert an Age header into the response. An Age header contains information about the age of the object, in seconds, as calculated by the integrated cache.

insertETag

Insert an ETag header in the response. With ETag header insertion, the integrated cache does not serve full responses on repeat requests.

cacheControl

Insert a Cache-Control header into the response.

quickAbortSize

If the size of an object that is being downloaded is less than or equal to the quick abort value, and a client aborts during the download, the cache stops downloading the response. If the object is larger than the quick abort size, the cache continues to download the response.

minResSize

Minimum size of a response that can be cached in this content group.

Default minimum response size is 0.

maxResSize

Maximum size of a response that can be cached in this content group.

memUsage

Current memory usage.

memDUsage

Current disk memory usage.

diskLimit

Maximum amount of disk that the cache can use. The effective limit is based on the available memory of the NetScaler appliance.

memLimit

Maximum amount of memory that the cache can use. The effective limit is based on the available memory of the NetScaler appliance.

ignoreReqCachingHdrs

Ignore Cache-Control and Pragma headers in the incoming request.

cacheNon304Hits

Cache non 304 hits.

cache304Hits

Cache 304 hits.

cacheCells

Number of cells.

cacheGroupIncarnation

Cache group incarnation.

minHits

Number of hits that qualifies a response for storage in this content group.

alwaysEvalPolicies

Force policy evaluation for each response arriving from the origin server. Cannot be set to YES if the Prefetch parameter is also set to YES.

persist

Setting persist to YES causes IC to save objects in contentgroup to disk.

persistHA

Setting persistHA to YES causes IC to save objects in contentgroup to Secondary node in HA deployment.

pinned

Do not flush objects from this content group under memory pressure.

lazyDnsResolve

Perform DNS resolution for responses only if the destination IP address in the request does not match the destination IP address of the cached response.

hitSelector

Selector for evaluating whether an object gets stored in a particular content group. A selector is an abstraction for a collection of PIXL expressions.

invalidSelector

Selector for invalidating objects in the content group. A selector is an abstraction for a collection of PIXL expressions.

policyName

Active cache policies referring to this group.

cacheNumInvalPolicy

Number of active Invalidation policies referring to this group.

markerCells

Numbers of marker cells in this group.

builtin

devno

count

stateflag

expire cache contentGroup

Forces expiration of all the objects in the specified content group. The next request for any object in the group is sent to the origin server.

Synopsys

expire cache contentGroup <name>

Arguments

name

Name of the content group whose objects are to be expired.

flush cache contentGroup

Flush the objects in the specified content group.

Synopsys

flush cache contentGroup <name> [-query <string> | -selectorValue <string>] [-host <string>]

Arguments

name

Name of the content group from which to flush objects, or "all" to flush all content groups.

query

Query string specifying individual objects to flush from this group by using parameterized invalidation. If this parameter is not set, all objects are flushed from the group.

host

Flush only objects that belong to the specified host. Do not use except with parameterized invalidation. Also, the Invalidation Restricted to Host parameter for the group must be set to YES.

selectorValue

Value of the selector to be used for flushing objects from the content group. Requires that an invalidation selector be configured for the content group.

stat cache contentGroup

Displays a summary of cache group statistics.

Synopsys

stat cache contentGroup [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

name

Name of the cache contentgroup for which to display statistics. If you do not set this parameter, statistics are shown for all cache contentgroups.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

non304 Hits for Content group (non304hit)

Non304 hits for ContentGroup

304 Hits for Content group (304hit)

304 hits for ContentGroup

Number of objects in contentgroup (cell)

Number of objects in contentgroup

Number of marker objects in contentgroup (Mrkcell)

Number of marker objects in contentgroup

Number of times contentgroup is flushed (flushed)

Number of times contentgroup is flushed

current memory usage (CurMem)

current memory usage

maximum memory usage limit (MaxMem)

maximum memory usage limit

Example

```
stat cache contentgroup
```

save cache contentGroup

Save the objects in the specified content group.

Synopsys

```
save cache contentGroup <name> [-tosecondary ( YES | NO )]
```

Arguments

name

The name of the content group whose objects are to be save.

tosecondary

content group whose objects are to be sent to secondary.

Possible values: YES, NO

Default value: NO

cache forwardProxy

The following operations can be performed on "cache forwardProxy":

[add](#) | [rm](#) | [show](#)

add cache forwardProxy

Allows the cache to act as a forward proxy for other NetScaler appliances or cache servers.

Synopsys

```
add cache forwardProxy <IPAddress> <port>
```

Arguments

IPAddress

IP address of the NetScaler appliance or a cache server for which the cache acts as a proxy. Requests coming to the NetScaler with the configured IP address are forwarded to the particular address, without involving the Integrated Cache in any way.

port

Port on the NetScaler appliance or a server for which the cache acts as a proxy

Minimum value: 1

rm cache forwardProxy

Removes the forward proxy address from the Integrated Cache. The cache does not act as a proxy to the specified IP address.

Synopsys

```
rm cache forwardProxy <IPAddress> <port>
```

Arguments

IPAddress

IP address of the NetScaler appliance or a server for which the cache was as a proxy.

port

Port on the NetScaler appliance or a server for which the cache acts as a proxy

Minimum value: 1

show cache forwardProxy

Displays the IP address and the corresponding ports for which the cache acted as a forward proxy.

Synopsys

```
show cache forwardProxy
```

Outputs

IPAddress

IP address of the NetScaler appliance or a cache server for which the cache acts as a proxy. Requests coming to the NetScaler with the configured IP address are forwarded to the particular address, without involving the Integrated Cache in any way.

port

Forward proxy port.

devno

count

stateflag

cache global

The following operations can be performed on "cache global":

[bind](#) | [unbind](#) | [show](#)

bind cache global

Binds the cache policy to one of the two global bind points (an unnamed policy label invoked at request time and an unnamed policy label invoked at the response time). The flow type of the policy implicitly determines which label it gets bound to. A policy becomes active only when it is bound. A globally bound policy, it is available to all virtual servers on the NetScaler appliance. All HTTP traffic is evaluated against the global policy labels. Each label contains an ordered list ordered by policies' priority values.

Synopsys

```
bind cache global <policy> -priority <positive_integer> [-gotoPriorityExpression <expression>] [-type <type>] [-invoke  
(<labelType> <labelName>)]
```

Arguments

policy

Name of the policy to bind. (A policy must be created before it can be bound.)

priority

Priority to assign to the policy. The appliance might disallow some priority values, depending on what you have already configured. For example, a response cache policy cannot have a higher priority than a request cache policy. Priority helps in dictating the order of policy evaluation.

Minimum value: 1

Maximum value: 2147483647

gotoPriorityExpression

Expression or other value specifying the priority of the next policy to evaluate if the current policy rule evaluates to TRUE. Specify one of the following values:

- * END. Terminate evaluation of this policy bank. This setting is equivalent to omitting the parameter.
- * NEXT. Evaluate the policy with the next higher priority.
- * An expression whose evaluation results in a number.

Evaluation of an expression determines the next action as follows:

- * If the expression evaluates to a priority number larger than the highest priority number in the policy bank, the next policy bank is evaluated.
- * If the expression evaluates to the priority of a policy with a lower priority (higher number) within the same policy bank, that policy is evaluated next.
- * If the expression evaluates to the priority of the current policy, the policy with the next-lower priority is evaluated next.

Any of the following results trigger an UNDEF condition

- * The expression cannot be evaluated.
- * The expression evaluates to a number that is smaller than the current policy's priority number

type

Bind point, specifying where to bind the policy.

Possible values: REQ_OVERRIDE, REQ_DEFAULT, RES_OVERRIDE, RES_DEFAULT

invoke

Invoke policies bound to a virtual server or a user-defined policy label. After the invoked policies are evaluated, the flow returns to the policy with the next priority. Applicable only to default-syntax policies.

labelType

Type of policy label to invoke.

Possible values: reqvserver, resvserver, policylabel

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE. (To invoke a label associated with a virtual server, specify the name of the virtual server.)

unbind cache global

Deactivate the policy by unbinding it from a global bind point.

Synopsys

unbind cache global <policy> [-type <type>] [-priority <positive_integer>]

Arguments

policy

Name of the policy to unbind.

type

Bind point from which to unbind the policy.

Possible values: REQ_OVERRIDE, REQ_DEFAULT, RES_OVERRIDE, RES_DEFAULT

priority

Priority of the NOPOLICY to be unbound. Required only you want to unbind a NOPOLICY that might have been bound to this policy label.

Minimum value: 1

Maximum value: 2147483647

show cache global

Displays the global bindings for cache policies.

Synopsys

show cache global [-type <type>]

Arguments

type

The bind point to which policy is bound. When you specify the type, detailed information about that bind point appears.

Possible values: REQ_OVERRIDE, REQ_DEFAULT, RES_OVERRIDE, RES_DEFAULT

Outputs

policyName

Name of the cache policy.

policy

Name of the cache policy.

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

invoke

Invoke policies bound to a virtual server or a user-defined policy label. After the invoked policies are evaluated, the flow returns to the policy with the next priority. Applicable only to default-syntax policies.

labelType

Type of policy label to invoke.

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE. (To invoke a label associated with a virtual server, specify the name of the virtual server.)

numpol

The number of policies bound to the bindpoint.

flowType

flowtype of the bound cache policy.

rule

The request/response rule that will trigger the given action.

action

The integrated cache action to be applied when the system sees content that matches the rules.

storeInGroup

The content group to store the object when the action directive is CACHE.

invalidGroups

The content group(s) to be invalidated when the action directive is INVALID.

invalidObjects

The content group(s) whose objects will be invalidated when the action directive is INVALID.

hits

Hits.

flags

Flags.

precedeDefRules

Override the default request/response cacheability rules.

stateflag

devno

count

Example

show cache global

cache object

The following operations can be performed on "cache object":

[show](#) | [expire](#) | [flush](#) | [save](#)

show cache object

Displays a list of all cached objects. The list displays the unique locator ID of each cached object along with the content group in which it was cached, and other details. To view more details of a specific cached object, use the -locator parameter along with this command.

Synopsys

```
show cache object [{-url <URL> (-host <string> [-port <port>] [-groupName <string>] [-httpMethod ( GET | POST )])}
| -locator <positive_integer> | -httpStatus <positive_integer> | -group <string> | -ignoreMarkerObjects ( ON | OFF ) | -
includeNotReadyObjects ( ON | OFF )]
```

Arguments

url

URL of the particular object whose details is required. Parameter "host" must be specified along with the URL.

locator

ID of the cached object.

Minimum value: 0

httpStatus

HTTP status of the object.

Minimum value: 0

host

Host name of the object. Parameter "url" must be specified.

port

Host port of the object. You must also set the Host parameter.

Default value: 80

Minimum value: 1

groupName

Name of the content group to which the object belongs. It will display only the objects belonging to the specified content group. You must also set the Host parameter.

httpMethod

HTTP request method that caused the object to be stored.

Possible values: GET, POST

Default value: GET

group

Name of the content group whose objects should be listed.

ignoreMarkerObjects

Ignore marker objects. Marker objects are created when a response exceeds the maximum or minimum response size for the content group or has not yet received the minimum number of hits for the content group.

Possible values: ON, OFF

includeNotReadyObjects

Include responses that have not yet reached a minimum number of hits before being cached.

Possible values: ON, OFF

Outputs

cacheResSize

Cache response size of the object.

cacheResHdrSize

Cache response header size of the object.

cacheETag

Cache ETag of the object.

httpStatusOutput

HTTP status of the object.

cacheResLastMod

Value of "Last-modified" header.

cacheControl

Cache-Control header of the object.

cacheResDate

Value of "Date" header

contentGroup

Name of the contentgroup in which it is stored.

destIP

Destination IP.

destIPV46

Destination IP.

destPort

Destination Port.

cacheCellComplex

The state of the parameterized caching on this cell.

hitParams

Parameterized hit evaluation of an object.

hitValues

Values of hitparams for this object.

cacheCellReqTime

Required time of the cache cell object.

cacheCellResTime

Response time to the cache cell object.

cacheCurAge

Current age of the cache object.

cacheCellExpires

Expiry time of the cache cell object in seconds.

cacheCellExpiresMilliSec

Expiry time of the cache cell object in milliseconds.

flushed

Specifies whether the object is flushed.

prefetch

Specifies whether Integrated Cache should attempt to refresh an object immediately before it goes stale.

prefetchPeriod

The duration in seconds of the period during which prefetch should be attempted, immediately before the object's calculated expiry time.

prefetchPeriodMilliSec

The duration in milliseconds of the period during which prefetch should be attempted, immediately before the object's calculated expiry time.

cacheCellCurReaders

Current readers of the cache cell object.

cacheCellCurMisses

Current misses of the cache cell object.

cacheCellHits

Cache cell hits.

cacheCellMisses

Cache cell misses.

cacheCellDHits

Cache cell disk hits.

cacheCellGzipCompressed

The state of the response being gzip-compressed.

cacheCellDeflateCompressed

The state of the response being deflate-compressed.

cacheCellCompressionFormat

Compression format of this object. Identity means not compressed

cacheCellAppFWMetadataExists

AppFirewall cache object.

cacheCellHttp11

The state of the response to be HTTP/1.1.

cacheCellWeakEtag

The state of the weak HTTP Entity Tag in the cell.

cacheCellResBadSize

The marked state of the cell.

markerReason

Reason for marking the cell.

cacheCellPollEveryTime

The state to poll every time on object.

cacheCellEtagInserted

The state of the ETag to be inserted by IC for this object.

cacheCellReadyWithLastByte

The state of the complete arrived response.

cacheInMemory

The cache data is available in memory.

cacheInDisk

The cache data is available in disk.

cacheInSecondary

The cache data is available in secondary in HA deployment.

cacheDirname

The directory name used if saved.

cacheFilename

The filename used if saved.

cacheCellDestipVerified

The state of DNS verification.

cacheCellFwpxyObj

The state of the object to be stored on a request to a forward proxy.

cacheCellBasefile

The state of delta being used as a basefile.

cacheCellMinHitFlag

The state of the minhit feature on this cell.

cacheCellMinHit

Min hit value for the object.

policy

Policy info for the object.

policyName

Policy which created the object.

selectorName

The hit selector for the object.

rule

Selectors for this object.

selectorValue

The HTTP request method that caused the object to be stored.

cacheUrls

List of cache object URLs.

numurls

Total number of cache object entries returned in cacheUrls field

warnBucketSkip

Bucket skipped warning.

totalObjs

Total objects.

httpCalloutCell

Is it a http callout cell ?

httpCalloutName

Name of the http callout

returnType

Return type of the http callout

httpCalloutResult

First few bytes of http callout response

locatorshow

ID of the cached object.

ceflags

Indicates state and type of cached cell

devno

count

stateflag

expire cache object

Forces expiry of a cached object. You have to specify the locator ID of the cached object by using the -locator parameter.

Synopsys

expire cache object (-locator <positive_integer> | (-url <URL> (-host <string> [-port <port>] [-groupName <string>] [-httpMethod (GET | POST)]))))

Arguments

locator

ID of the cached object to be expired To view the locator ID of the cached objects, use the show cache object command.

Minimum value: 0

url

The URL of the object to be expired.

host

The host of the object to be expired.

port

The host port of the object to be expired.

Default value: 80

Minimum value: 1

groupName

Name of the content group to which the object belongs.

httpMethod

HTTP request method that caused the object to be stored.

Possible values: GET, POST

Default value: GET

flush cache object

Removes a cached object from memory and from disk (if it has a disk copy). You have to specify the locator ID of the cached object by using the -locator parameter

Synopsys

flush cache object (-locator <positive_integer> | (-url <URL> (-host <string> [-port <port>] [-groupName <string>] [-httpMethod (GET | POST)])) [-force]

Arguments

locator

ID of the cached object. To view the locator ID of the cached objects, use the show cache object command.

Minimum value: 0

url

URL of the object to be flushed. You must also set the Host parameter.

host

Host of the object to be flushed. Must provide the "url" parameter along with the host.

port

Host port of the object to be flushed. Must provide the "host" parameter along with the port.

Default value: 80

Minimum value: 1

groupName

Name of the content group to which the object belongs. Must provide the "\\host\\" parameter along with the group name.

httpMethod

HTTP request method that caused the object to be stored. All objects cached by that method will be flushed.

Possible values: GET, POST

Default value: GET

force

Force all copies to be flushed including on disk.

save cache object

Save a cached object to local disk.

Synopsys

save cache object [-locator <positive_integer>] [-tosecondary (YES | NO)]

Arguments

locator

The ID of the cached object.

Minimum value: 0

tosecondary

Object will be saved onto Secondary.

Possible values: YES, NO

Default value: NO

cache parameter

The following operations can be performed on "cache parameter":

[set](#) | [unset](#) | [show](#)

set cache parameter

Modifies the global configuration of the integrated cache. You can modify the settings of various parameters.

Synopsys

```
set cache parameter [-memLimit <MBytes>] [-via <string>] [-verifyUsing <verifyUsing>] [-maxPostLen  
<positive_integer>] [-prefetchMaxPending <positive_integer>] [-enableBypass ( YES | NO )] [-undefAction ( NOCACHE | RESET )] [-enableHaObjPersist ( YES | NO )]
```

Arguments

memLimit

Amount of memory available for storing the cache objects. In practice, the amount of memory available for caching can be less than half the total memory of the NetScaler appliance.

via

String to include in the Via header. A Via header is inserted into all responses served from a content group if its Insert Via flag is set.

verifyUsing

Criteria for deciding whether a cached object can be served for an incoming HTTP request. Available settings function as follows:

HOSTNAME - The URL, host name, and host port values in the incoming HTTP request header must match the cache policy. The IP address and the TCP port of the destination host are not evaluated. Do not use the HOSTNAME setting unless you are certain that no rogue client can access a rogue server through the cache.

HOSTNAME_AND_IP - The URL, host name, host port in the incoming HTTP request header, and the IP address and TCP port of

the destination server, must match the cache policy.

DNS - The URL, host name and host port in the incoming HTTP request, and the TCP port must match the cache policy. The host name is used for DNS lookup of the destination server's IP address, and is compared with the set of addresses returned by the DNS lookup.

Possible values: HOSTNAME, HOSTNAME_AND_IP, DNS

maxPostLen

Maximum number of POST body bytes to consider when evaluating parameters for a content group for which you have configured hit parameters and invalidation parameters.

Default value: 4096

Minimum value: 0

Maximum value: 131072

prefetchMaxPending

Maximum number of outstanding prefetches in the Integrated Cache.

Minimum value: 0

enableBypass

Evaluate the request-time policies before attempting hit selection. If set to NO, an incoming request for which a matching object is found in cache storage results in a response regardless of the policy configuration.

If the request matches a policy with a NOCACHE action, the request bypasses all cache processing.

This parameter does not affect processing of requests that match any invalidation policy.

Possible values: YES, NO

undefAction

Action to take when a policy cannot be evaluated.

Possible values: NOCACHE, RESET

enableHaObjPersist

The HA object persisting parameter. When this value is set to YES, cache objects can be synced to Secondary in a HA deployment. If set to NO, objects will never be synced to Secondary node.

Possible values: YES, NO

Default value: NO

unset cache parameter

Use this command to remove cache parameter settings. Refer to the set cache parameter command for meanings of the arguments.

Synopsis

```
unset cache parameter [-memLimit] [-via] [-verifyUsing] [-maxPostLen] [-prefetchMaxPending] [-enableBypass] [-undefAction] [-enableHaObjPersist]
```

show cache parameter

Displays the global configuration of the Integrated Cache.

Synopsis

```
show cache parameter
```

Outputs

diskLimit

The disk limit for the Integrated Cache.

maxDiskLimit

The maximum value of the memory limit for the Integrated Cache.

memLimit

The memory limit for the Integrated Cache.

memLimitActive

Active value of the memory limit for the Integrated Cache.

maxMemLimit

The maximum value of the memory limit for the Integrated Cache.

via

The string that is inserted in the "Via" header.

verifyUsing

The criteria for deciding whether a cached object can be served for an incoming HTTP request.

maxPostLen

The maximum POST body size that the IC can accumulate.

prefetchCur

Number of current outstanding prefetches in the IC.

prefetchMaxPending

The maximum number of outstanding prefetches on the content group.

enableBypass

When this value is set to NO, an incoming request will serve a hit if a matching object is found in cache storage, regardless of the cacheability policy configuration. If set to YES, the bound request cacheability policies are evaluated before attempting any hit selection in the cache storage. If the request matches a policy with a NOCACHE action, the request will bypass all cache processing. This flag does not affect processing of requests that match any invalidation policy.

undefAction

Action to take when a policy cannot be evaluated.

enableHaObjPersist

The HA object persisting parameter. When this value is set to YES, cache objects can be synced to Secondary in a HA deployment. If set to NO, objects will never be synced to Secondary node.

cache policy

The following operations can be performed on "cache policy":

add | **rm** | **set** | **unset** | **show** | **stat** | **rename**

add cache policy

Creates an integrated caching policy. The newly created policy is in inactive state. To activate the policy, use the bind cache global command.

Synopsis

```
add cache policy <policyName> -rule <expression> -action <action> [-storeInGroup <string>] [-invalidGroups <string> ...] [-invalidObjects <string> ...] [-undefAction ( NOCACHE | RESET )]
```

Arguments

policyName

Name for the policy. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Can be changed after the policy is created.

rule

Expression against which the traffic is evaluated.

Note:

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

The following requirements apply only to the NetScaler CLI:

- * If the expression includes one or more spaces, enclose the entire expression in double quotation marks.
- * If the expression itself includes double quotation marks, escape the quotations by using the \ character.
- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Action to apply to content that matches the policy.

- * CACHE or MAY_CACHE action - positive cachability policy
- * NOCACHE or MAY_NOCACHE action - negative cachability policy
- * INVALID action - Dynamic Invalidation Policy

Possible values: CACHE, NOCACHE, MAY_CACHE, MAY_NOCACHE, INVALID

storeInGroup

Name of the content group in which to store the object when the final result of policy evaluation is CACHE. The content group must exist before being mentioned here. Use the "show cache contentgroup" command to view the list of existing content groups.

invalidGroups

Content group(s) to be invalidated when the INVALID action is applied. Maximum number of content groups that can be specified is 16.

invalidObjects

Content groups(s) in which the objects will be invalidated if the action is INVALID.

undefAction

Action to be performed when the result of rule evaluation is undefined.

Possible values: NOCACHE, RESET

rm cache policy

Removes the specified caching policy. Make sure that the policy is not bound globally or to a virtual server. A bound policy cannot be removed.

Synopsis

```
rm cache policy <policyName>
```

Arguments

policyName

Name of the cache policy to be removed.

set cache policy

Modifies the specified attributes of an existing cache policy. The rule, flow type, can be changed only if action and undefAction (if present) are of NEUTRAL flow type.

Synopsis

```
set cache policy <policyName> [-rule <expression>] [-action <action>] [-storeInGroup <string>] [-invalGroups <string> ...] [-invalObjects <string> ...] [-undefAction ( NOCACHE | RESET )]
```

Arguments

policyName

Name for the policy. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Can be changed after the policy is created.

rule

Expression against which the traffic is evaluated.

Note:

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

The following requirements apply only to the NetScaler CLI:

- * If the expression includes one or more spaces, enclose the entire expression in double quotation marks.
- * If the expression itself includes double quotation marks, escape the quotations by using the \ character.
- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Action to apply to content that matches the policy.

- * CACHE or MAY_CACHE action - positive cachability policy
- * NOCACHE or MAY_NOCACHE action - negative cachability policy

* INVALID action - Dynamic Invalidation Policy

Possible values: CACHE, NOCACHE, MAY_CACHE, MAY_NOCACHE, INVALID

storeInGroup

Name of the content group in which to store the object when the final result of policy evaluation is CACHE. The content group must exist before being mentioned here. Use the "show cache contentgroup" command to view the list of existing content groups.

invalGroups

Content group(s) to be invalidated when the INVALID action is applied. Maximum number of content groups that can be specified is 16.

invalObjects

Content groups(s) in which the objects will be invalidated if the action is INVALID.

undefAction

Action to be performed when the result of rule evaluation is undefined.

Possible values: NOCACHE, RESET

Example

```
set cache policy pol9 -rule "http.req.HEADER(\\\\"header\\\\" ).CONTAINS(\\\\"qh2\\\\" )" "
```

unset cache policy

Use this command to remove cache policy settings. Refer to the set cache policy command for meanings of the arguments.

Synopsis

```
unset cache policy <policyName> [-storeInGroup] [-invalGroups] [-invalObjects] [-undefAction]
```

show cache policy

Displays all configured cache policies. To display details about a particular cache policy, specify the name of the policy. When all caching policies are displayed, the order of the displayed policies within each group is the same as the evaluation order of the policies. There are three groups: request policies, response policies, and dynamic invalidation policies.

Synopsis

```
show cache policy [<policyName>] show cache policy stats - alias for 'stat cache policy'
```

Arguments

policyName

Name of the cache policy about which to display details.

Outputs

stateflag

rule

The request/response rule that will trigger the specified action.

action

The integrated cache action to be applied when the system sees content that matches the rules.

storeInGroup

The content group that will store the object when the action directive is CACHE.

invalGroups

The content group(s) to be invalidated when the action directive is INVALID.

invalObjects

The content group(s) whose objects will be invalidated when the action directive is INVALID.

priority

Priority.

hits

Hits.

undefAction

A CACHE action, to be used by the policy when the rule evaluation turns out to be undefined.

undefHits

Number of Undef hits.

flags

Flag.

precedeDefRules

Override default request/response cacheability rules.

activePolicy

Indicates whether policy is bound or not.

boundTo

Location where policy is bound

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

labelType

Type of policy label invocation.

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

bindPolicyType**vserverType****builtin****devno****count**

stat cache policy

Displays a summary of cache policy statistics.

Synopsys

```
stat cache policy [<policyName>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

policyName

Name of the cache policy for which to display statistics. If you do not set this parameter, statistics are shown for all cache policies.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Policy hits (Hits)

Number of hits on the policy

Policy undef hits (Undefhits)

Number of undef hits on the policy

Example

```
stat cache policy
```

rename cache policy

Renames an existing cache policy.

Synopsys

rename cache policy <policyName>@ <newName>@

Arguments

policyName

Existing name of the cache policy.

newName

New name for the cache policy. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

Example

```
rename cache policy oldname newname
```

cache policylabel

The following operations can be performed on "cache policylabel":

[add](#) | [rm](#) | [bind](#) | [unbind](#) | [show](#) | [stat](#) | [rename](#)

add cache policylabel

Creates a user-defined cache policy label. A policy label is a bind point of a group of policies.

Synopsis

```
add cache policylabel <labelName> -evaluates <evaluates>
```

Arguments

labelName

Name for the label. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Can be changed after the label is created.

evaluates

When to evaluate policies bound to this label: request-time or response-time.

Possible values: REQ, RES, MSSQL_REQ, MSSQL_RES, MYSQL_REQ, MYSQL_RES

Example

```
add cache policylabel cache_http_url -evaluates REQ
```

rm cache policylabel

Removes the specified integrated caching policy label.

Synopsis

```
rm cache policylabel <labelName>
```

Arguments

labelName

Name of the label to be removed.

Example

```
rm cache policylabel cache_http_url
```

bind cache policylabel

Binds a cache policy to a policy label.

Synopsis

```
bind cache policylabel <labelName> -policyName <string> -priority <positive_integer> [-gotoPriorityExpression <expression>] [-invoke (<labelType> <labelName>)]
```

Arguments

labelName

Name of the policy label to invoke if the current policy rule evaluates to TRUE.

policyName

Name of the cache policy to bind to the policy label.

priority

Integer specifying the priority of this policy within the policy label. A lower number specifies a higher priority. The policies bound to the label are evaluated in the order of their priorities.

Minimum value: 1

Maximum value: 2147483647

gotoPriorityExpression

Expression or other value specifying the priority of the next policy, within the policy label, to evaluate if the current policy evaluates to TRUE. Specify one of the following values:

- * NEXT - Evaluate the policy with the next higher numbered priority.
- * END - Stop evaluation.
- * USE_INVOCATION_RESULT - Applicable if this policy invokes another policy label. If the final goto in the invoked policy label has a value of END, the evaluation stops. If the final goto is anything other than END, the current policy label performs a NEXT.
- * An expression that evaluates to a number.

If you specify an expression, its evaluation result determines the next policy to evaluate, as follows:

- * If the expression evaluates to a higher numbered priority, that policy is evaluated next.
- * If the expression evaluates to the priority of the current policy, the policy with the next higher priority number is evaluated next.
- * If the expression evaluates to a priority number that is numerically higher than the highest priority number, policy evaluation ends.

An UNDEF event is triggered if:

- * The expression is invalid.
- * The expression evaluates to a priority number that is numerically lower than the current policy's priority.
- * The expression evaluates to a priority number that is between the current policy's priority number (say, 30) and the highest priority number (say, 100), but does not match any configured priority number (for example, the expression evaluates to the number 85). This example assumes that the priority number increments by 10 for every successive policy, and therefore a priority number of 85 does not exist in the policy label.

invoke

Invoke policies bound to a virtual server or a user-defined policy label. After the invoked policies are evaluated, the flow returns to the policy with the next-lower priority.

labelType

Type of policy label to invoke: an unnamed label associated with a virtual server, or user-defined policy label.

Possible values: reqvserver, resvserver, policylabel

Example

```
i) bind cache policylabel cache_http_url pol_1 1 2 -invoke reqvserver CURRENT ii) bind c
```

unbind cache policylabel

Unbinds a policy from a cache-policy label.

Synopsys

unbind cache policylabel <labelName> -policyName <string> [-priority <positive_integer>]

Arguments

labelName

Name of the cache policy label from which to unbind the policy.

policyName

Name of the policy to unbind from the label.

priority

Required only if you want to unbind a NOPOLICY that might have been bound to this policy label.

Minimum value: 1

Maximum value: 2147483647

Example

```
unbind cache policylabel cache_http_url pol_1
```

show cache policylabel

Displays information about all cache-policy labels or about the specified cache-policy label.

Synopsys

show cache policylabel [<labelName>]

Arguments

labelName

Name of the cache-policy label about which to display information.

Outputs

stateflag

flags

evaluates

When to evaluate policies bound to this label: request-time or response-time.

numpol

Number of polices bound to label.

hits

Number of times policy label was invoked.

policyName

Name of the cache policy to bind to the policy label.

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

invoke

Invoke policies bound to a virtual server or a user-defined policy label. After the invoked policies are evaluated, the flow returns to the policy with the next-lower priority.

labelType

Type of policy label to invoke: an unnamed label associated with a virtual server, or user-defined policy label.

labelName

Name of the policy label to invoke if the current policy rule evaluates to TRUE.

flowType

Flowtype of the bound cache policy.

builtin

devno

count

Example

```
i) show cache policylabel cache_http_url ii) show cache policylabel
```

stat cache policylabel

Displays statistics of cache policy label(s).

Synopsys

```
stat cache policylabel [<labelName>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

labelName

Name of the cache-policy label for which to display statistics. If you do not set this parameter statistics are shown for all cache-policy labels.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Policy Label Hits (Hits)

Number of times policy label was invoked.

rename cache policylabel

Renames a cache-policy label.

Synopsys

```
rename cache policylabel <labelName>@ <newName>@
```

Arguments

labelName

Existing name of the cache-policy label.

newName

New name for the cache-policy label. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

Example

```
rename cache policylabel oldname newname
```

cache selector

The following operations can be performed on "cache selector":

[add](#) | [rm](#) | [set](#) | [show](#)

add cache selector

Creates an Integrated Cache selector. A selector is an abstraction for a collection of PIXL expressions. After creating a selector, you can use it as a hit selector, invalidation selector, or both. You must specify at least one expression when you create a selector.

Synopsis

```
add cache selector <selectorName> <rule> ...
```

Arguments

selectorName

Name for the selector. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

rule

One or multiple PIXL expressions for evaluating an HTTP request or response.

rm cache selector

Removes cache selectors. Note: A selector being used as a hit or invalidation selector in any content group cannot be removed without unsetting it from the content group.

Synopsis

```
rm cache selector <selectorName>
```

Arguments

selectorName

Name of the selector.

set cache selector

Modify the set of PIXL expressions associated with a cache selector.

Synopsis

```
set cache selector <selectorName> <rule> ...
```

Arguments

selectorName

Name of the selector to be modified.

rule

One or multiple PIXL expressions for evaluating an HTTP request or response.

show cache selector

Displays all cache selectors, or the specified.

Synopsys

show cache selector [<selectorName>]

Arguments

selectorName

Name of the selector to display.

Outputs

flags

Flags.

rule

Rule.

devno

count

stateflag

cache stats

The following operations can be performed on "cache stats":

show cache stats

show cache stats is an alias for stat cache

Synopsys

show cache stats - alias for 'stat cache'

CLI Commands

The entities on which you can perform NetScaler CLI operations:

- o alias
- o backup
- o batch
- o cli attribute
- o cli mode
- o cli prompt
- o cls
- o config
- o exit
- o help
- o history
- o man
- o quit
- o source
- o unalias
- o whoami

alias

The following operations can be performed on "alias":

alias

Create (short) aliases for (long) commands. Aliases are saved across NSCLI sessions. If no argument is specified, the alias command will display existing aliases.

Synopsys

```
alias [<pattern> [(command)]]
```

Arguments

pattern

Alias name. (Can be a regular expression.)

command

Target command

Example

```
alias info "show ns info"
```

backup

The following operations can be performed on "backup":

backup

backup cache object to local disk

Synopsys

backup -pattern <string>

Arguments

pattern

Name of the alias

Example

backup cache object -locator <id>

batch

The following operations can be performed on "batch":

batch

Use this command to read the contents of a file and execute each line as a separate CLI command. Each command in the file must be on a separate line. Lines starting with # are considered comments.

Synopsys

```
batch -fileName <input_filename> [-outfile <output_filename>] [-ntimes <positive_integer>]
```

Arguments

fileName

The name of the batch file.

outfile

The name of the file where the executed batch file will write its output. The default is standard output.

ntimes

The number of times the batch file will be executed.

Default value: 1

Minimum value: 0

Example

```
batch -f cmds.txt
```

cli attribute

The following operations can be performed on "cli attribute":

show cli attribute

Display attributes of the NetScaler CLI

Synopsys

show cli attribute

Outputs

qquote

The construct that is used to quote strings that are to be taken as-is, without interpreting escape sequences like "

".

This construct consists of: a 'q', followed by a delimiter character; the string follows immediately after the delimiter and is terminated by the first matching delimiter character. (The set of possible delimiter characters is listed below.)

For example, q/a

/ will result in a three-character string ('a', '/', 'n'); whereas "a

" results in a two-character string ('a' followed by a newline).

qquoteDelimiters

The set of characters that can be used as the delimiter in a q// construct. Characters shown in pairs must be used that way, whereas characters shown singly serve as their own matching delimiter.

For example, q?abc? and q{abc} are valid q// constructs, and evaluate to the string "abc"; q{abc{ is however not a valid q// construct so it will evaluate to the string "q{abc{".

cli mode

The following operations can be performed on "cli mode":

[set](#) | [unset](#) | [show](#)

set cli mode

Use this command to specify how the CLI should display command output.

Synopsys

```
set cli mode [-page ( ON | OFF )] [-total ( ON | OFF )] [-color ( ON | OFF )] [-disabledFeatureAction  
<disabledFeatureAction>] [-timeout <secs>] [-timeoutKind <timeoutKind>] [-regex ( ON | OFF )]
```

Arguments

page

Determines whether output that spans more than one screen is "paged". Specify ON to pause the display after each screen of output.

Possible values: ON, OFF

Default value: OFF

total

Determines whether CLI "show" commands display a total count of objects before displaying the objects themselves.

Possible values: ON, OFF

Default value: OFF

color

Specifies whether output can be shown in color, if the terminal supports it.

Possible values: ON, OFF

Default value: OFF

disabledFeatureAction

Specifies what will happen when a configuration command is issued for a disabled feature. The following values are allowed:

NONE - The action is allowed, and no warning message is issued.;

ALLOW - The action is allowed, but a warning message is issued.;

DENY - The action is not allowed.;

HIDE - Commands that configure disabled features are hidden, and the CLI behaves as if they did not exist.

Possible values: NONE, ALLOW, DENY, HIDE

Default value: NS_ALLOW

timeout

CLI session inactivity timeout, in seconds. If Restrictedtimeout argument of system parameter is enabled, Timeout can have values in the range [300-86400] seconds. If Restrictedtimeout argument of system parameter is disabled, Timeout can have values in the range [0, 10-100000000] seconds. Default value is 900 seconds.

Default value: -1

timeoutKind

From where the timeout has been inherited.

Possible values: User, Group, Global, Climode

regex

If ON, regular expressions can be used as argument values

Possible values: ON, OFF

Default value: ON

unset cli mode

Use this command to remove cli mode settings. Refer to the set cli mode command for meanings of the arguments.

Synopsys

unset cli mode [-page] [-total] [-color] [-disabledFeatureAction] [-timeout] [-timeoutKind] [-regex]

show cli mode

Use this command to display the current settings of parameters that can be set with the 'set cli mode' command.

Synopsys

show cli mode

Outputs

page

Determines whether output that spans more than one screen is "paged". Specify ON to pause the display after each screen of output.

total

Determines whether CLI "show" commands display a total count of objects before displaying the objects themselves.

color

Specifies whether output can be shown in color, if the terminal supports it.

disabledFeatureAction

Specifies what will happen when a configuration command is issued for a disabled feature. The following values are allowed:

NONE - The action is allowed, and no warning message is issued.;

ALLOW - The action is allowed, but a warning message is issued.;

DENY - The action is not allowed.;

HIDE - Commands that configure disabled features are hidden, and the CLI behaves as if they did not exist.

argMark

mark

noLicenseAction

no licence

diagLevel

diagnostic level

timeout

CLI session inactivity timeout, in seconds. If Restrictedtimeout argument of system parameter is enabled, Timeout can have values in the range [300-86400] seconds. If Restrictedtimeout argument of system parameter is disabled, Timeout can have values in the range [0, 10-100000000] seconds. Default value is 900 seconds.

timeoutKind

From where the timeout has been inherited.

regex

If ON, regular expressions can be used as argument values

r

regular expression

format

format

stats**serverPort**

cli prompt

The following operations can be performed on "cli prompt":

[clear](#) | [set](#) | [show](#)

clear cli prompt

Use this command to return the CLI prompt to the default (a single '>').

Synopsys

clear cli prompt

set cli prompt

Use this command to customize the CLI prompt.

Synopsys

set cli prompt <promptString>

Arguments

promptString

The prompt string. The following special values are allowed:

%! - will be replaced by the history event number

%u - will be replaced by the NetScaler user name

%h - will be replaced by the NetScaler hostname

%t - will be replaced by the current time

%T - will be replaced by the current time (24 hr format)

%d - will be replaced by the current date

%s - will be replaced by the node state

Example

```
> set cli prompt "%h %T" Done lb-ns1 15:16>
```

show cli prompt

Use this command to display the current CLI prompt, with special values like '%h' unexpanded.

Synopsys

show cli prompt

Outputs

promptString

Example

```
10.101.4.22 15:20> sh cli prompt CLI prompt is set to "%h %T" Done
```

cls

The following operations can be performed on "cls":

cls

Clear the screen and reposition cursor at top right.

Synopsys

cls

config

The following operations can be performed on "config":

config

Enter this command to enter contextual mode.

Synopsys

config

exit

The following operations can be performed on "exit":

exit

Use this command to back out one level in config mode, or to terminate the CLI when not in config mode.);

Synopsys

exit

help

The following operations can be performed on "help":

help

Use this command to display help information for a CLI command, for a group of commands, or for all CLI commands.

Synopsys

help [(commandName) | <groupName> | -all]

Arguments

commandName

The name of a command for which you want full usage information.

groupName

The name of a command group for which you want basic usage information.

all

Use this option to request basic usage information for all commands.

Example

1.To view help information for adding a virtual server, enter the following CLI command: 1

history

The following operations can be performed on "history":

history

Use this command to see the history of the commands executed on CLI.

Synopsys

history

Example

```
history                                1 add snmp trap SPECIFIC 10.102.130.228
```


man

The following operations can be performed on "man":

man

Use this command to invoke the man page for the specified command. You can specify the command in full, or partially, if it is uniquely resolvable.

Synopsys

`man [(commandName)]`

Arguments

commandName

The name of the command.

Example

`man add vs`

quit

The following operations can be performed on "quit":

quit

Use this command to terminate the CLI. Note: typing <Ctrl>+<d> will also terminate the CLI.

Synopsys

quit

source

The following operations can be performed on "source":

source

Use this command to read the contents of a file and execute each line as a separate CLI command. Each command in the file being read must be on a separate line. Lines starting with # are considered comments.

Synopsys

```
source <fileName>
```

Arguments

fileName

The name of the file to be sourced.

Example

```
source cmds.txt
```

unalias

The following operations can be performed on "unalias":

unalias

Remove an alias

Synopsys

unalias <pattern>

Arguments

pattern

Name of the alias

Example

```
unalias info
```

whoami

The following operations can be performed on "whoami":

whoami

Show the current user.

Synopsys

whoami

Outputs

userName

loggedIn

Cluster Commands

The entities on which you can perform NetScaler CLI operations:

- o cluster
- o cluster files
- o cluster instance
- o cluster node
- o cluster nodegroup
- o cluster sync

cluster

The following operations can be performed on "cluster":

join cluster

Joins the appliance to the cluster. You must execute this command from the NetScaler IP (NSIP) address of the node that you want to add to the cluster. This command is the second part of the two-step process of adding a cluster node. The first part is adding this node to the cluster by using the add cluster node command from the cluster IP address. This operation is not permitted if any node in the cluster is in the Sync state.

Synopsys

```
join cluster -clip <ip_addr> {-password }
```

Arguments

clip

Cluster IP address to which to add the node.

password

Password for the nsroot account of the configuration coordinator (CCO).

cluster files

The following operations can be performed on "cluster files":

sync cluster files

Synchronizes SSL Certificates, SSL CRL lists, SSL VPN bookmarks, and other files from the configuration coordinator (CCO) to the other cluster nodes. Execute this command from the cluster IP address only. This command is automatically triggered from the CCO when a new node is added to a cluster and periodically triggered to synchronize updated files between the cluster nodes. Note: Files on non-CCO nodes are not deleted if they do not exist on the CCO.

Synopsys

sync cluster files [<Mode> ...]

Arguments

Mode

The directories and files to be synchronized. The available settings function as follows:

Mode	Paths
------	-------

all	/nsconfig/ssl/
-----	----------------

	/var/netScaler/ssl/
--	---------------------

	/var/vpn/bookmark/
--	--------------------

	/nsconfig/dns/
--	----------------

	/nsconfig/htmlinjection/
--	--------------------------

	/netScaler/htmlinjection/ens/
--	-------------------------------

	/nsconfig/monitors/
--	---------------------

	/nsconfig/nstemplates/
--	------------------------

	/nsconfig/ssh/
--	----------------

	/nsconfig/rc.netScaler
--	------------------------

	/nsconfig/resolv.conf
--	-----------------------

	/nsconfig/inetd.conf
--	----------------------

	/nsconfig/syslog.conf
--	-----------------------

	/nsconfig/snmpd.conf
--	----------------------

	/nsconfig/ntp.conf
--	--------------------

	/nsconfig/httpd.conf
--	----------------------

	/nsconfig/sshd_config
--	-----------------------

	/nsconfig/hosts
--	-----------------

	/nsconfig/enckey
--	------------------

	/var/nslw.bin/etc/krb5.conf
--	-----------------------------

	/var/nslw.bin/etc/krb5.keytab
--	-------------------------------

	/var/lib/likewise/db/
--	-----------------------

/var/download/
/var/wi/tomcat/webapps/
/var/wi/tomcat/conf/Catalina/localhost/
/var/wi/java_home/lib/security/cacerts
/var/wi/java_home/jre/lib/security/cacerts
/var/netscaler/locdb/
ssl /nsconfig/ssl/
/var/netscaler/ssl/
bookmarks /var/vpn/bookmark/
dns /nsconfig/dns/
htmlinjection /nsconfig/htmlinjection/
imports /var/download/
misc /nsconfig/license/
/nsconfig/rc.conf
all_plus_misc Includes *all* files and /nsconfig/license/ and /nsconfig/rc.conf.
Default value: all

Example

```
sync cluster files ssl or sync cluster files all
```

cluster instance

The following operations can be performed on "cluster instance":

add | **rm** | **set** | **unset** | **enable** | **disable** | **show** | **stat**

add cluster instance

Adds a cluster instance to the appliance. Execute this command on only the first node that you add to the cluster.

Synopsys

```
add cluster instance <cld> [-deadInterval <secs>] [-helloInterval <msecs>] [-preemption ( ENABLED | DISABLED )]  
[-quorumType ( MAJORITY | NONE )]
```

Arguments

cld

Unique number that identifies the cluster.

Minimum value: 1

Maximum value: 16

deadInterval

Amount of time, in seconds, after which nodes that do not respond to the heartbeats are assumed to be down.

Default value: 3

Minimum value: 3

Maximum value: 60

helloInterval

Interval, in milliseconds, at which heartbeats are sent to each cluster node to check the health status.

Default value: 200

Minimum value: 200

Maximum value: 1000

preemption

Preempt a cluster node that is configured as a SPARE if an ACTIVE node becomes available.

Possible values: ENABLED, DISABLED

Default value: DISABLED

quorumType

Quorum Configuration Choices - "Majority" (recommended) requires majority of nodes to be online for the cluster to be UP. "None" relaxes this requirement.

Possible values: MAJORITY, NONE

Default value: MAJORITY

Example

```
add cluster instance 1
```

rm cluster instance

Removes the cluster instance from the node. You must execute this command on the NetScaler IP (NSIP) address of the node.

Synopsys

```
rm cluster instance <cld>
```

Arguments

cld

Unique number that identifies the cluster.

Minimum value: 1

Maximum value: 16

Example

```
rm cluster instance 1
```

set cluster instance

Modifies the specified attributes of a cluster instance.

Synopsys

```
set cluster instance <cld> [-deadInterval <secs>] [-helloInterval <msecs>] [-preemption ( ENABLED | DISABLED )] [-quorumType ( MAJORITY | NONE )]
```

Arguments

cld

ID of the cluster instance to be modified.

Minimum value: 1

Maximum value: 16

deadInterval

Amount of time, in seconds, after which nodes that do not respond to the heartbeats are assumed to be down.

Default value: 3

Minimum value: 3

Maximum value: 60

helloInterval

Interval, in milliseconds, at which heartbeats are sent to each cluster node to check the health status.

Default value: 200

Minimum value: 200

Maximum value: 1000

preemption

Preempt a cluster node that is configured as a SPARE if an ACTIVE node becomes available.

Possible values: ENABLED, DISABLED

Default value: DISABLED

quorumType

Quorum Configuration Choices - "Majority" (recommended) requires majority of nodes to be online for the cluster to be UP. "None" relaxes this requirement.

Possible values: MAJORITY, NONE

Default value: MAJORITY

Example

```
set cluster instance 1 -preemption ENABLED
```

unset cluster instance

Use this command to remove cluster instance settings. Refer to the set cluster instance command for meanings of the arguments.

Synopsis

```
unset cluster instance <cld> [-deadInterval] [-helloInterval] [-preemption] [-quorumType]
```

enable cluster instance

Enables a cluster instance.

Synopsis

```
enable cluster instance <cld>
```

Arguments

cld

ID of the cluster instance that you want to enable.

Minimum value: 1

Maximum value: 16

Example

```
enable cluster instance 1
```

disable cluster instance

Disables a cluster instance.

Synopsis

```
disable cluster instance <cld>
```

Arguments

cld

ID of the cluster instance that you want to disable.

Minimum value: 1

Maximum value: 16

Example

```
disable cluster instance 1
```

show cluster instance

Displays information about the cluster instance and its nodes.

Synopsys

```
show cluster instance [<cld>]
```

Arguments

cld

Unique number that identifies the cluster.

Minimum value: 1

Maximum value: 16

Outputs

deadInterval

Amount of time, in seconds, after which nodes that do not respond to the heartbeats are assumed to be down.

helloInterval

Interval, in milliseconds, at which heartbeats are sent to each cluster node to check the health status.

preemption

Preempt a cluster node that is configured as a SPARE if an ACTIVE node becomes available.

adminstate

Cluster Admin State.

quorumType

Quorum Configuration Choices - "Majority" (recommended) requires majority of nodes to be online for the cluster to be UP. "None" relaxes this requirement.

propState

Enable/Disable the execution of commands on the cluster. This will not impact the execution of commands on individual cluster nodes by using the NSIP.

nodeId

The unique number that identifies a cluster.

IPAddress

The IP Address of the node.

flags

The flags for this entry.

masterState

Master state.

health

Node Health state.

clusterHealth

Node clusterd state.

effectiveState

Node effective health state.

state

Active, Spare or Passive. An active node serves traffic. A spare node serves as a backup for active nodes. A passive node does not serve traffic. This may be useful during temporary maintenance activity where it is desirable that the node takes part in the consensus protocol, but not serve traffic.

flag

Cluster Flag.

operationalstate

Cluster Operational State.

status

Cluster Operational State.

isConfigurationCoordinator

This argument is used to determine whether the node is configuration coordinator (CCO).

isLocalnode

This argument is used to determine whether it is local node.

RSSKeyMismatch

This argument is used to determine if there is a RSS key mismatch at cluster instance level.

LicenseMismatch

This argument is used to determine if there is a License mismatch at cluster instance level.

JumboNotSupported

This argument is used to determine if Jumbo framework is not supported at cluster instance level.

NodeRSSKeyMismatch

This argument is used to determine if there is a RSS key mismatch at cluster node level.

NodeLicenseMismatch

This argument is used to determine if there is a License mismatch at cluster node level.

NodeJumboNotSupported

This argument is used to determine if Jumbo framework not supported at cluster node level.

stateflag

State Flag.

operationalPropState

Cluster Operational Propagation State.

devno

count

Example

An example of the command's output is as follows: 1)Cluster ID: 1 Dead Interval: :

stat cluster instance

Displays statistics for a cluster instance.

Synopsys

stat cluster instance [<cld>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats
(basic | full)]

Arguments

cld

ID of the cluster instance for which to display statistics.

Minimum value: 1

Maximum value: 16

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Cluster size (CLNumNodes)

Number of nodes in the cluster.

Cluster status (CLCurEnable)

State of the cluster.

Configuration coordinator (CLViewLeader)

NSIP address of the Configuration Coordinator of the cluster.

Total steered packets (TotSteeredPkts)

Total number of packets steered on the cluster backplane.

Traffic received (Bkplane Rx)

Traffic received on backplane (in mbits)

Traffic transmitted (Bkplane Tx)

Traffic transmitted from backplane (in mbits)

Dropped steered packets (DFDdropPkts)

Number of steered packets that are dropped.

Propagation timeout (propTimeout)

Number of times the update to the client timed-out.

cluster node

The following operations can be performed on "cluster node":

add | **set** | **unset** | **rm** | **show** | **stat**

add cluster node

Adds a NetScaler appliance to a cluster.

Synopsys

```
add cluster node <nodeId>@ <IPAddress>@ [-state <state>] [-backplane <interface_name>@] [-priority <positive_integer>]
```

Arguments

nodeId

Unique number that identifies the cluster node.

Minimum value: 0

Maximum value: 31

IPAddress

NetScaler IP (NSIP) address of the appliance to add to the cluster. Must be an IPv4 address.

state

Admin state of the cluster node. The available settings function as follows:

ACTIVE - The node serves traffic.

SPARE - The node does not serve traffic unless an ACTIVE node goes down.

PASSIVE - The node does not serve traffic, unless you change its state. PASSIVE state is useful during temporary maintenance activities in which you want the node to take part in the consensus protocol but not to serve traffic.

Possible values: ACTIVE, SPARE, PASSIVE

Default value: PASSIVE

backplane

Interface through which the node communicates with the other nodes in the cluster. Must be specified in the three-tuple form n/c/u, where n represents the node ID and c/u refers to the interface on the appliance.

Minimum value: 1

priority

Preference for selecting a node as the configuration coordinator. The node with the lowest priority value is selected as the configuration coordinator.

When the current configuration coordinator goes down, the node with the next lowest priority is made the new configuration coordinator. When the original node comes back up, it will preempt the new configuration coordinator and take over as the configuration coordinator.

Note: When priority is not configured for any of the nodes or if multiple nodes have the same priority, the cluster elects one of the nodes as the configuration coordinator.

Default value: 31

Minimum value: 0

Maximum value: 31

Example

```
add cluster node 1 1.1.1.1 -backplane 1/1/1 -state ACTIVE
```

set cluster node

Modifies the attributes of a cluster node.

Synopsys

```
set cluster node <nodeId>@ [-state <state>] [-backplane <interface_name>@] [-priority <positive_integer>]
```

Arguments

nodeId

ID of the cluster node to be modified.

Minimum value: 0

Maximum value: 31

state

Admin state of the cluster node. The available settings function as follows:

ACTIVE - The node serves traffic.

SPARE - The node does not serve traffic unless an ACTIVE node goes down.

PASSIVE - The node does not serve traffic, unless you change its state. PASSIVE state is useful during temporary maintenance activities in which you want the node to take part in the consensus protocol but not to serve traffic.

Possible values: ACTIVE, SPARE, PASSIVE

Default value: PASSIVE

backplane

Interface through which the node communicates with the other nodes in the cluster. Must be specified in the three-tuple form n/c/u, where n represents the node ID and c/u refers to the interface on the appliance.

Minimum value: 1

priority

Preference for selecting a node as the configuration coordinator. The node with the lowest priority value is selected as the configuration coordinator.

When the current configuration coordinator goes down, the node with the next lowest priority is made the new configuration coordinator. When the original node comes back up, it will preempt the new configuration coordinator and take over as the configuration coordinator.

Note: When priority is not configured for any of the nodes or if multiple nodes have the same priority, the cluster elects one of the nodes as the configuration coordinator.

Default value: 31

Minimum value: 0

Maximum value: 31

Example

```
set cluster node 1 -state PASSIVE
```

unset cluster node

Use this command to remove cluster node settings. Refer to the set cluster node command for meanings of the arguments.

Synopsys

```
unset cluster node <nodeId>@ [-state] [-backplane] [-priority]
```

rm cluster node

Removes a node from the cluster and removes the cluster instance from the node. You must execute this command on the cluster IP address.

Synopsys

```
rm cluster node <nodeId>
```

Arguments

nodeId

ID of the cluster node to be removed from the cluster.

Minimum value: 0

Maximum value: 31

Example

```
rm cluster node 1
```

show cluster node

Displays information about the cluster node.

Synopsys

```
show cluster node [<nodeId>@]
```

Arguments

nodeId

ID of the cluster node for which to display information. If an ID is not provided, information about all nodes is shown.

Default value: 255

Minimum value: 0

Maximum value: 31

Outputs

IPAddress

The IP Address of the node.

flags

The flags for this entry.

clusterHealth

Node clusterd state.

effectiveState

Node effective health state.

operationalSyncState

Node Operational Reconciliation state.

masterState

Node Master state.

health

Node Health state.

state

Active, Spare or Passive.

backplane

Interface through which the node communicates with the other nodes in the cluster. Must be specified in the three-tuple form n/c/u, where n represents the node ID and c/u refers to the interface on the appliance.

priority

Preference for selecting a node as the configuration coordinator. The node with the lowest priority value is selected as the configuration coordinator.

When the current configuration coordinator goes down, the node with the next lowest priority is made the new configuration coordinator. When the original node comes back up, it will preempt the new configuration coordinator and take over as the configuration coordinator.

Note: When priority is not configured for any of the nodes or if multiple nodes have the same priority, the cluster elects one of the nodes as the configuration coordinator.

isConfigurationCoordinator

This argument is used to determine whether the node is configuration coordinator (CCO).

isLocalnode

This argument is used to determine whether it is local node.

NodeRSSKeyMismatch

This argument is used to determine if there is a RSS key mismatch at cluster node level.

NodeLicenseMismatch

This argument is used to determine if there is a License mismatch at cluster node level.

NodeJumboNotSupported

This argument is used to determine if Jumbo framework not supported at cluster node level.

stateflag**nodeList**

Nodelist for displaying Heartbeat not seen interfaces on a cluster node

ifacesList

Interface list corresponding to nodelist for Heartbeat not seen interfaces on a cluster node

enabledIfaces

Enabled Interfaces on a cluster node.

disabledIfaces

Disabled Interfaces on a cluster node.

partialFailIfaces

Partial Failure Interfaces on a cluster node.

hamonIfaces

Hamon Interfaces on a cluster node.

ifaces

Interfaces status on cluster node.

devno**count**

Example

An example of the command's output is as follows: 1 cluster node: 1)Node ID: 1

stat cluster node

Displays statistics for a cluster node.

Synopsys

```
stat cluster node [<nodeId>@] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

nodeId

ID of the cluster node for which to display statistics. If an ID is not provided, statistics are shown for all nodes.

Minimum value: 0

Maximum value: 31

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Sync state (Sync State)

Sync state of the cluster node.

Health

Health of the cluster node.

Node IP (NodeIP)

NSIP address of the cluster node.

Operational state (OpState)

Operational state of the cluster node.

Heartbeats transmitted (HB Sent)

Number of heartbeats sent. When executed from the NSIP address, shows the statistics for local node only. For remote node it shows a value of 0. When executed from the cluster IP address, shows all the statistics.

Heartbeats received (HB Rcvd)

Number of heartbeats received. When executed from the NSIP address, shows the statistics for local node only. For remote node it shows a value of 0. When executed from the cluster IP address, shows all the statistics.

Current node-node connections (NNMCurConn)

Number of connections open for node-to-node communication.

Node-node messages transmitted (NNMTotConnTx)

Number of node-to-node messages sent. When executed from the NSIP address, shows the statistics for local node only. For remote node it shows a value of 0. When executed from the cluster IP address, shows all the statistics.

Node-node messages received (NNMTotConnRx)

Number of node-to-node messages received. When executed from the NSIP address, shows the statistics for local node only. For remote node it shows a value of 0. When executed from the cluster IP address, shows all the statistics.

PTP operational state (PTP State)

PTP state of the node. This state is Master for one node and Slave for the rest. When executed from the NSIP address, shows the statistics for local node only. For remote node it shows UNKNOWN. When executed from the cluster IP address, shows all the statistics.

PTP packets transmitted (PTP Tx)

Number of PTP packets transmitted by the node. When executed from the NSIP address, shows the statistics for local node only. For remote node it shows a value of 0. When executed from the cluster IP address, shows all the statistics.

PTP packets received (PTP Rx)

Number of PTP packets received on the node. When executed from the NSIP address, shows the statistics for local node only. For remote node it shows a value of 0. When executed from the cluster IP address, shows all the statistics.

Multicast/Broadcast send errors (NNMErrMsend)

Number of errors in sending node-to-node multicast/broadcast messages. When executed from the NSIP address, shows the statistics for local node only. For remote node it shows a value of 0. When executed from the cluster IP address, shows all the statistics.

(Health)

Health of the node in the cluster.

CH State

Health State of the node with respect to sync in the cluster.

cluster nodegroup

The following operations can be performed on "cluster nodegroup":

add | **show** | **set** | **unset** | **bind** | **unbind** | **rm**

add cluster nodegroup

Adds a nodegroup to the cluster. A nodegroup is a set of cluster nodes to which entities can be bound. Entities that are bound to a specific nodegroup are active on all the nodes of the group and not active on the nodes that are not part of the group.

Synopsis

```
add cluster nodegroup <name>@ [-strict ( YES | NO )] [-sticky ( YES | NO )]
```

Arguments

name

Name of the nodegroup. The name uniquely identifies the nodegroup on the cluster.

strict

Specifies whether cluster nodes, that are not part of the nodegroup, will be used as backup for the nodegroup.

* Enabled - When one of the nodes goes down, no other cluster node is picked up to replace it. When the node comes up, it will continue being part of the nodegroup.

* Disabled - When one of the nodes goes down, a non-nodegroup cluster node is picked up and acts as part of the nodegroup. When the original node of the nodegroup comes up, the backup node will be replaced.

Possible values: YES, NO

Default value: NO

sticky

Only one node can be bound to nodegroup with this option enabled. It specifies whether to preempt the traffic for the entities bound to nodegroup when owner node goes down and rejoins the cluster.

* Enabled - When owner node goes down, backup node will become the owner node and takes the traffic for the entities bound to the nodegroup. When bound node rejoins the cluster, traffic for the entities bound to nodegroup will not be steered back to this bound node. Current owner will have the ownership till it goes down.

* Disabled - When one of the nodes goes down, a non-nodegroup cluster node is picked up and acts as part of the nodegroup. When the original node of the nodegroup comes up, the backup node will be replaced.

Possible values: YES, NO

Default value: NO

Example

```
add cluster nodegroup ng1 -strict yes
```

show cluster nodegroup

Displays information about the available nodegroups.

Synopsis

```
show cluster nodegroup [<name>]
```


Arguments

name

Name of the nodegroup to be displayed. If a name is not provided, information about all nodegroups is displayed.

Outputs

node

Nodes in the nodegroup

strict

Specifies whether cluster nodes, that are not part of the nodegroup, will be used as backup for the nodegroup.

* Enabled - When one of the nodes goes down, no other cluster node is picked up to replace it. When the node comes up, it will continue being part of the nodegroup.

* Disabled - When one of the nodes goes down, a non-nodegroup cluster node is picked up and acts as part of the nodegroup. When the original node of the nodegroup comes up, the backup node will be replaced.

sticky

Only one node can be bound to nodegroup with this option enabled. It specifies whether to preempt the traffic for the entities bound to nodegroup when owner node goes down and rejoins the cluster.

* Enabled - When owner node goes down, backup node will become the owner node and takes the traffic for the entities bound to the nodegroup. When bound node rejoins the cluster, traffic for the entities bound to nodegroup will not be steered back to this bound node. Current owner will have the ownership till it goes down.

* Disabled - When one of the nodes goes down, a non-nodegroup cluster node is picked up and acts as part of the nodegroup. When the original node of the nodegroup comes up, the backup node will be replaced.

vServer

vserver that need to be bound to this nodegroup.

currentNodeMask

Bitmap of current nodes in this nodegroup

backupNodeMask

Bitmap of backup nodes in this nodegroup

boundedEntitiesCntFromPE

Count of bounded entities to this nodegroup according to PE

activeList

Active node list of this nodegroup

backupList

Backup node list of this nodegroup

identifierName

stream identifier and rate limit identifier that need to be bound to this nodegroup.

gslbSite

vserver that need to be bound to this nodegroup.

service

name of the service bound to this nodegroup.

stateflag

devno

count

set cluster nodegroup

Modifies the attributes of a cluster nodegroup.

Synopsys

```
set cluster nodegroup <name>@ [-strict ( YES | NO )]
```

Arguments

name

Name of the nodegroup to be modified.

strict

Specifies whether cluster nodes, that are not part of the nodegroup, will be used as backup for the nodegroup.

* Enabled - When one of the nodes goes down, no other cluster node is picked up to replace it. When the node comes up, it will continue being part of the nodegroup.

* Disabled - When one of the nodes goes down, a non-nodegroup cluster node is picked up and acts as part of the nodegroup. When the original node of the nodegroup comes up, the backup node will be replaced.

Possible values: YES, NO

Default value: NO

Example

```
set cluster nodegroup ng1 -strict yes
```

unset cluster nodegroup

Unset nodes from the given nodegroup or unset strict option. Refer to the set cluster nodegroup command for meanings of the arguments.

Synopsys

```
unset cluster nodegroup <name>@ [-strict]
```

Example

```
unset cluster nodegroup ng1 -strict
```

bind cluster nodegroup

Binds a cluster node or an entity to the given nodegroup. A node can be bound to more than one nodegroup.

Synopsys

```
bind cluster nodegroup <name> (-node <positive_integer>@ | -vServer <string> | -identifierName <string> | -gslbSite <string> | -service <string>)
```

Arguments

name

Name of the nodegroup to which you want to bind a cluster node or an entity.

node

ID of the node to be bound to the nodegroup.

Default value: -1

Minimum value: 0

Maximum value: 31

vServer

Name of the virtual server to be bound to the nodegroup.

identifierName

Name of stream or limit identifier to be bound to the nodegroup.

gslbSite

Name of the GSLB site to be unbound from the nodegroup.

service

Name of the service to be unbound from the nodegroup.

Example

```
bind cluster nodegroup ng1 -vserver v1
```

unbind cluster nodegroup

Unbinds a cluster node or an entity from a given nodegroup.

Synopsys

```
unbind cluster nodegroup <name> (-node <positive_integer>@ | -vServer <string> | -identifierName <string> | -gslbSite <string> | -service <string>)
```

Arguments

name

Name of the nodegroup from which you want to unbind a cluster node or an entity.

node

ID of the node to be unbound from the nodegroup.

Default value: -1

Minimum value: 0

Maximum value: 31

vServer

Name of the virtual server to be unbound from the nodegroup.

identifierName

Name of stream or limit identifier to be unbound from the nodegroup.

gslbSite

Name of the GSLB site to be unbound from the nodegroup.

service

Name of the service to be unbound from the nodegroup.

Example

```
unbind cluster nodegroup ng1 -vserver v1
```

rm cluster nodegroup

Removes a nodegroup from the cluster.

Synopsys

```
rm cluster nodegroup <name>@
```

Arguments

name

Name of the nodegroup to be removed.

Example

```
rm cluster nodegroup ng1
```

cluster sync

The following operations can be performed on "cluster sync":

force cluster sync

Synchronize the configurations of a cluster node from the configuration coordinator (CCO). This command must be executed from the NSIP of the node that is to be synchronized.

Synopsys

```
force cluster sync
```

Example

```
force cluster sync
```

Compression Commands

The entities on which you can perform NetScaler CLI operations:

- o `cmp`
- o `cmp action`
- o `cmp global`
- o `cmp parameter`
- o `cmp policy`
- o `cmp policylabel`
- o `cmp stats`

cmp

The following operations can be performed on "cmp":

stat cmp

Display compression statistics.

Synopsys

```
stat cmp [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

Bandwidth saving (%) (DIBndSav)

Bandwidth saving from delta compression expressed as percentage.

Delta compression ratio (DICmpRt)

Ratio of compressible data received to compressed data transmitted.If this ratio is one (uncmp:1.0) that means compression is disabled or we are not able to compress even a single compressible packet.

Decompression ratio (DTCmpRt)

Ratio of decompressed data transmitted to compressed data received.

Bandwidth saving (%) (DBndSav)

Bandwidth saving from TCP compression expressed as percentage.

TCP compression ratio (TCmpRt)

Ratio of compressible data received to compressed data transmitted.If this ratio is one (uncmp:1.0) that means compression is disabled or we are not able to compress even a single compressible packet.

TCP Bandwidth saving (%) (BndSav)

Bandwidth saving from TCP compression expressed as percentage.

Total HTTP compression ratio

Ratio of total HTTP data received to total HTTP data transmitted.

HTTP Bandwidth saving (%) (HttpBndSav)

Bandwidth saving from TCP compression expressed as percentage.

HTTP compression ratio

Ratio of the compressible data received from the server to the compressed data sent to the client.

HTTP compression requests

Number of HTTP compression requests the NetScaler receives for which the response is successfully compressed. For example, after you enable compression and configure services, if you send requests to the NetScaler with the following header information: ?Accept-Encoding: gzip, deflate?, and NetScaler compresses the corresponding response, this counter is incremented.

Compressible bytes received

Number of bytes that can be compressed, which the NetScaler receives from the server. This gives the content length of the response that the NetScaler receives from server.

Compressed bytes transmitted

Number of bytes the NetScaler sends to the client after compressing the response from the server.

Compressible packets received

Number of HTTP packets that can be compressed, which the NetScaler receives from the server.

Compressed packets transmitted

Number of HTTP packets that the NetScaler sends to the client after compressing the response from the server.

Compressible bytes received (TCmpRxB)

Number of bytes that can be compressed, which the NetScaler receives from the server. This gives the content length of the response that the NetScaler receives from server.

Compressible packets received (TCmpRxP)

Total number of compressible packets received by NetScaler.

Compressed bytes transmitted (TCmpTxB)

Number of bytes that the NetScaler sends to the client after compressing the response from the server.

Compressed packets transmitted (TCmpTxP)

Number of TCP packets that the NetScaler sends to the client after compressing the response from the server.

Quantum compression (TCmpQuan)

Number of times the NetScaler compresses a quantum of data. NetScaler buffers the data received from the server till it reaches the quantum size and then compresses the buffered data and transmits to the client.

Push flag compression (TCmpPush)

Number of times the NetScaler compresses data on receiving a TCP PUSH flag from the server. The PUSH flag ensures that data is compressed immediately without waiting for the buffered data size to reach the quantum size.

End Of Input compression (TCmpEoi)

Number of times the NetScaler compresses data on receiving End Of Input (FIN packet). When the NetScaler receives End Of Input (FIN packet), it compresses the buffered data immediately without waiting for the buffered data size to reach the quantum size.

Timer compression (TCmpTmr)

Number of times the NetScaler compresses data on expiration of data accumulation timer. The timer expires if the server response is very slow and consequently, the NetScaler does not receive response for a certain amount of time. Under such a condition, the NetScaler compresses the buffered data immediately without waiting for the buffered data size to reach the quantum size.

Compressed bytes received (DCmpTRxB)

Total number of compressed bytes received by NetScaler.

Compressed packets received (DCmpTRxP)

Total number of compressed packets received by NetScaler.

Decompressed bytes transmitted (DCmpTTxB)

Total number of decompressed bytes transmitted by NetScaler.

Decompressed packets transmitted (DCmpTTxP)

Total number of decompressed packets transmitted by NetScaler.

Wrong data (DCmpErrD)

Number of data errors encountered while decompressing.

Less Data (DCmpErrL)

Number of times NetScaler received less data than declared by protocol.

More Data (DCmpErrM)

Number of times NetScaler received more data than declared by protocol.

Memory failures (DCmpMem)

Number of times memory failures occurred while decompressing.

Unknown (DCmpErrU)

Number of times unknown errors occurred while decompressing.

Delta compression requests (DICmpRx)

Total number of delta compression requests received by NetScaler.

Delta compression applied (DIDone)

Total number of delta compressions done by NetScaler.

Compressible bytes received (DICmpRxB)

Total number of delta-compressible bytes received by NetScaler.

Compressed bytes transmitted (DICmpTxB)

Total number of delta-compressed bytes transmitted by NetScaler.

First-time access (DICmpFAc)

Total number of delta compression first accesses.

Compressible packets received (DICmpRxP)

Number of delta-compressible packets received.

Compressed packets transmitted (DICmpTxP)

Total number of delta-compressed packets transmitted by NetScaler.

Basefile requests served (DICBSrv)

Total number of basefile requests served by NetScaler.

Basefile bytes transmitted (DICBTxB)

Number of basefile bytes transmitted by NetScaler.

Delta compression bypassed (DICmpEBy)

Number of times delta-compression bypassed by NetScaler.

Basefile write header failed (DICmpEBW)

Number of times basefile could not be updated in NetScaler cache.

Basefile no-store miss (DICmpENM)

Number of times basefile was not found in NetScaler cache.

Request information too big (DICmpERB)

Number of times basefile request URL was too large.

Request info alloc failed (DICmpERF)

Number of times requested basefile could not be allocated.

Session allocation failed (DICmpESF)

Number of times delta compression session could not be allocated.

Response bytes received (HTRspbRx)

Total number of bytes of HTTP response data received.

cmp action

The following operations can be performed on "cmp action":

[add](#) | [rm](#) | [show](#) | [set](#) | [unset](#) | [rename](#)

add cmp action

Creates a compression action. Note: User-defined compression actions supplement the built-in compression actions. The built-in compression actions, NOCOMPRESS, COMPRESS, GZIP, and DEFLATE, are always available. Available settings functions as follows: * NOCOMPRESS - Disables compression for data that matches the associated policy. * COMPRESS - Enable GZIP or DEFLATE compression, depending on which is supported by the browser. * GZIP - Enable GZIP compression. For browsers that do not support GZIP, compression is disabled. * DEFLATE - Enable DEFLATE compression for a specific policy. For browsers that do not support DEFLATE, compression is disabled.

Synopsys

```
add cmp action <name> <cmpType> [-addVaryHeader <addVaryHeader> -varyHeaderValue <string>]
```

Arguments

name

Name of the compression action. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Can be changed after the action is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my cmp action" or 'my cmp action').

cmpType

Type of compression performed by this action.

Available settings function as follows:

* COMPRESS - Apply GZIP or DEFLATE compression to the response, depending on the request header. Prefer GZIP.

* GZIP - Apply GZIP compression.

* DEFLATE - Apply DEFLATE compression.

* NOCOMPRESS - Do not compress the response if the request matches a policy that uses this action.

Possible values: compress, gzip, deflate, nocompress

addVaryHeader

Control insertion of the Vary header in HTTP responses compressed by NetScaler. Intermediate caches store different versions of the response for different values of the headers present in the Vary response header.

Possible values: GLOBAL, DISABLED, ENABLED

Default value: GLOBAL

varyHeaderValue

The value of the HTTP Vary header for compressed responses.

Example

```
add cmp action nocmp NOCOMPRESS
```

rm cmp action

Removes the specified compression action.

Synopsys

```
rm cmp action <name>
```

Arguments

name

Name of the action to be removed.

Example

```
rm cmp action cmp_action_name
```

show cmp action

Displays information about all the built-in and user-defined compression actions, or detailed information about the specified action.

Synopsys

```
show cmp action [<name>]
```

Arguments

name

Name of the action for which to display detailed information.

Outputs

cmpType

Type of compression performed by this action.

Available settings function as follows:

* COMPRESS - Apply GZIP or DEFLATE compression to the response, depending on the request header. Prefer GZIP.

* GZIP - Apply GZIP compression.

* DEFLATE - Apply DEFLATE compression.

* NOCOMPRESS - Do not compress the response if the request matches a policy that uses this action.

deltaType

The type of delta action if compression type is delta compression.

addVaryHeader

Control insertion of the Vary header in HTTP responses compressed by NetScaler. Intermediate caches store different versions of the response for different values of the headers present in the Vary response header.

varyHeaderValue

The value of the HTTP Vary header for compressed responses.

stateflag

flags

builtin

Flag to determine whether compression is default or not

isDefault

A value of true is returned if it is a default policy

devno

count

Example

Example 1 The following example shows output from the `show cmp action` command when no cu:

set cmp action

Modifies the specified parameters of a compression action.

Synopsis

`set cmp action <name> [-cmpType <cmpType>] [-addVaryHeader <addVaryHeader> -varyHeaderValue <string>]`

Arguments

name

Name of the compression action. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Can be changed after the action is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my cmp action" or 'my cmp action').

cmpType

Type of compression performed by this action.

Available settings function as follows:

* COMPRESS - Apply GZIP or DEFLATE compression to the response, depending on the request header. Prefer GZIP.

* GZIP - Apply GZIP compression.

* DEFLATE - Apply DEFLATE compression.

* NOCOMPRESS - Do not compress the response if the request matches a policy that uses this action.

Possible values: compress, gzip, deflate, nocompress

addVaryHeader

Control insertion of the Vary header in HTTP responses compressed by NetScaler. Intermediate caches store different versions of the response for different values of the headers present in the Vary response header.

Possible values: GLOBAL, DISABLED, ENABLED

Default value: GLOBAL

varyHeaderValue

The value of the HTTP Vary header for compressed responses.

Example

```
set cmp action compact1 -addVaryHeader ENABLED -varyHeaderValue User-Agent
```

unset cmp action

Use this command to remove cmp action settings. Refer to the set cmp action command for meanings of the arguments.

Synopsis

```
unset cmp action <name> -addVaryHeader
```

rename cmp action

Renames a compression action.

Synopsis

```
rename cmp action <name>@ <newName>@
```

Arguments

name

Existing name of the action.

newName

New name for the compression action. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at

(@), equals (=), and hyphen (-) characters.

Choose a name that can be correlated with the function that the action performs.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my cmp action" or 'my cmp action').

Example

```
rename cmp policy oldname newname
```

cmp global

The following operations can be performed on "cmp global":

[bind](#) | [unbind](#) | [show](#)

bind cmp global

Binds (activates) the compression policy globally. Note that the compression feature requires a compression license. When you enable the compression feature, all of the built-in compression policies are bound globally.

Synopsys

```
bind cmp global <policyName> [-priority <positive_integer>] [-state ( ENABLED | DISABLED )] [-gotoPriorityExpression <expression>] [-type <type>] [-invoke (<labelType> <labelName>)]
```

Arguments

policyName

Name of the policy to bind globally.

priority

Positive integer specifying the priority of the policy. The lower the number, the higher the priority. By default, policies within a label are evaluated in the order of their priority numbers.

In the configuration utility, you can click the Priority field and edit the priority level or drag the entry to a new position in the list. If you drag the entry to a new position, the priority level is updated automatically.

Minimum value: 1

Maximum value: 2147483647

state

Operational state of the globally bound policy.

Possible values: ENABLED, DISABLED

Default value: ENABLED

gotoPriorityExpression

Expression or other value specifying the priority of the next policy, within the policy label, to evaluate if the current policy evaluates to TRUE. Specify one of the following values:

- * NEXT - Evaluate the policy with the next higher numbered priority.
- * END - Stop evaluation.
- * USE_INVOCATION_RESULT - Applicable if this policy invokes another policy label. If the final goto in the invoked policy label has a value of END, the evaluation stops. If the final goto is anything other than END, the current policy label performs a NEXT.
- * An expression that evaluates to a number.

If you specify an expression, its evaluation result determines the next policy to evaluate, as follows:

- * If the expression evaluates to a higher numbered priority, that policy is evaluated next.
- * If the expression evaluates to the priority of the current policy, the policy with the next higher priority number is evaluated next.
- * If the expression evaluates to a priority number that is numerically higher than the highest priority number, policy evaluation ends.

An UNDEF event is triggered if:

- * The expression is invalid.
- * The expression evaluates to a priority number that is numerically lower than the current policy's priority.
- * The expression evaluates to a priority number that is between the current policy's priority number (say, 30) and the highest priority number (say, 100), but does not match any configured priority number (for example, the expression evaluates to the number 85). This example assumes that the priority number increments by 10 for every successive policy, and therefore a priority number of 85 does not exist in the policy label.

type

Global bind point, specifying where to bind the policy. This is relevant for advanced (default-syntax) policies only.

Possible values: REQ_OVERRIDE, REQ_DEFAULT, RES_OVERRIDE, RES_DEFAULT, NONE

Default value: NONE

invoke

Invoke policies bound to a virtual server or a policy label. After the invoked policies are evaluated, the flow returns to the policy with the next priority. Applicable only for default-syntax policies.

labelType

Type of policy label invocation. This argument is relevant only for advanced (default-syntax) policies.

Possible values: reqvserver, resvserver, policylabel

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE. Applicable only to advanced (default-syntax) policies.

Example

```
add cmp policy pdf_cmp -rule "RES.HTTP.HEADER Content-Type CONTAINS application/pdf" -re:
```

unbind cmp global

Deactivates a globally bound HTTP compression policy.

Synopsys

```
unbind cmp global <policyName> [-type <type> [-priority <positive_integer>]]
```

Arguments

policyName

Name of the compression policy to unbind.

type

Bind point, specifying from where to unbind the policy. Applicable only to advanced (default-syntax) policies.

Possible values: REQ_OVERRIDE, REQ_DEFAULT, RES_OVERRIDE, RES_DEFAULT, NONE

priority

Integer specifying the priority of the policy. The lower the number, the higher the priority. By default, policies within a label are evaluated in the order of their priority numbers.

Note that in the configuration utility, you can click the Priority field and edit the priority level or drag the entry to a new position in the list. If you drag the entry to a new position, the priority level is updated automatically.

Minimum value: 1

Maximum value: 2147483647

Example

To view the globally active compression policies, enter the following command: `> show cmp`

show cmp global

Displays the globally bound HTTP compression policies.

Synopsys

`show cmp global [-type <type>]`

Arguments

type

Bind point to which the policy is bound.

Possible values: REQ_OVERRIDE, REQ_DEFAULT, RES_OVERRIDE, RES_DEFAULT

Outputs

stateflag

policyName

The name of the globally bound HTTP compression policy.

priority

Positive integer specifying the priority of the policy. The lower the number, the higher the priority. By default, policies within a label are evaluated in the order of their priority numbers.

In the configuration utility, you can click the Priority field and edit the priority level or drag the entry to a new position in the list. If you drag the entry to a new position, the priority level is updated automatically.

state

The current state of the policy binding. This attribute is relevant only for CLASSIC policies.

numpol

The number of policies bound to the bindpoint.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE

invoke

Invoke flag. This attribute is relevant only for ADVANCED policies

labelType

Type of policy label invocation.

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

policyType

Policy type (Classic/Advanced) to be bound.Used for display.

devno

count

Example

```
> show cmp global          4 Globally Active Compression Policies: 1)      Policy Name: n:
```

cmp parameter

The following operations can be performed on "cmp parameter":

[set](#) | [unset](#) | [show](#)

set cmp parameter

Configures the compression parameters.

Synopsis

```
set cmp parameter [-cmpLevel <cmpLevel>] [-quantumSize <positive_integer>] [-serverCmp ( ON | OFF )] [-minResSize <positive_integer>] [-cmpBypassPct <positive_integer>] [-cmpOnPush ( ENABLED | DISABLED )] [-policyType ( CLASSIC | ADVANCED )] [-addVaryHeader ( ENABLED | DISABLED )] [-varyHeaderValue <string>] [-externalCache ( YES | NO )]
```

Arguments

cmpLevel

Specify a compression level. Available settings function as follows:

- * Optimal - Corresponds to a gzip GZIP level of 5-7.
- * Best speed - Corresponds to a gzip level of 1.
- * Best compression - Corresponds to a gzip level of 9.

Possible values: optimal, bestspeed, bestcompression

Default value: optimal

quantumSize

Minimum quantum of data to be filled before compression begins.

Default value: 57344

Minimum value: 8

Maximum value: 63488

serverCmp

Allow the server to send compressed data to the NetScaler appliance. With the default setting, the NetScaler appliance handles all compression.

Possible values: ON, OFF

Default value: ON

minResSize

Smallest response size, in bytes, to be compressed.

Minimum value: 0

cmpBypassPct

NetScaler CPU threshold after which compression is not performed. Range: 0 - 100

Default value: 100

Minimum value: 0

Maximum value: 100

cmpOnPush

NetScaler appliance does not wait for the quantum to be filled before starting to compress data. Upon receipt of a packet with a PUSH flag, the appliance immediately begins compression of the accumulated packets.

Possible values: ENABLED, DISABLED

Default value: DISABLED

policyType

Type of policy. Available settings function as follows:

* Classic - Classic policies evaluate basic characteristics of traffic and other data.

* Advanced - Advanced policies (which have been renamed as default syntax policies) can perform the same type of evaluations as classic policies. They also enable you to analyze more data (for example, the body of an HTTP request) and to configure more operations in the policy rule (for example, transforming data in the body of a request into an HTTP header).

Possible values: CLASSIC, ADVANCED

Default value: CLASSIC

addVaryHeader

Control insertion of the Vary header in HTTP responses compressed by NetScaler. Intermediate caches store different versions of the response for different values of the headers present in the Vary response header.

Possible values: ENABLED, DISABLED

Default value: DISABLED

varyHeaderValue

The value of the HTTP Vary header for compressed responses. If this argument is not specified, a default value of "Accept-Encoding" will be used.

externalCache

Enable insertion of Cache-Control: private response directive to indicate response message is intended for a single user and must not be cached by a shared or proxy cache.

Possible values: YES, NO

Default value: NO

Example

```
set cmp param -cmpLevel bestspeed -quantumSize 20480
```

unset cmp parameter

Use this command to remove cmp parameter settings. Refer to the set cmp parameter command for meanings of the arguments.

Synopsys

```
unset cmp parameter [-cmpLevel] [-quantumSize] [-serverCmp] [-minResSize] [-cmpBypassPct] [-cmpOnPush] [-policyType] [-addVaryHeader] [-varyHeaderValue] [-externalCache]
```

show cmp parameter

Displays the values of the compression parameters. Example: > show cmp parameter Configured compression parameters: Compression level: optimal Quantum size: 4555 Server-side compression: ON Minimum HTTP response size for compression: 0 CPU load at which to bypass compression: 100% Compression on PUSH: DISABLED Compression policy type: CLASSIC Vary header insertion: DISABLED Disable external cache: NO

Synopsys

show cmp parameter

Outputs

cmpLevel

Specify a compression level. Available settings function as follows:

- * Optimal - Corresponds to a gzip GZIP level of 5-7.
- * Best speed - Corresponds to a gzip level of 1.
- * Best compression - Corresponds to a gzip level of 9.

quantumSize

Minimum quantum of data to be filled before compression begins.

serverCmp

Compression enabled/disabled at back-end server.

heurExpiry

Heuristic basefile expiry.

heurExpiryThres

Threshold compression ratio for heuristic basefile expiry, multiplied by 100. For example, to set the threshold ratio to 1.25, specify 125.

heurExpiryHistWt

For heuristic basefile expiry, weightage to be given to historical delta compression ratio, specified as percentage. For example, to give 25% weightage to historical ratio (and therefore 75% weightage to the ratio for current delta compression transaction), specify 25.

minResSize

Smallest response size, in bytes, to be compressed.

cmpBypassPct

NetScaler CPU threshold after which compression is not performed. Range: 0 - 100

cmpOnPush

NetScaler appliance does not wait for the quantum to be filled before starting to compress data. Upon receipt of a packet with a PUSH flag, the appliance immediately begins compression of the accumulated packets.

policyType

Type of policy. Available settings function as follows:

- * Classic - Classic policies evaluate basic characteristics of traffic and other data.
- * Advanced - Advanced policies (which have been renamed as default syntax policies) can perform the same type of evaluations as classic policies. They also enable you to analyze more data (for example, the body of an HTTP request) and to configure more operations in the policy rule (for example, transforming data in the body of a request into an HTTP header).

addVaryHeader

Control insertion of the Vary header in HTTP responses compressed by NetScaler. Intermediate caches store different versions of the response for different values of the headers present in the Vary response header.

varyHeaderValue

The value of the HTTP Vary header for compressed responses. If this argument is not specified, a default value of "Accept-Encoding" will be used.

externalCache

Enable insertion of Cache-Control: private response directive to indicate response message is intended for a single user and must not be cached by a shared or proxy cache.

cmp policy

The following operations can be performed on "cmp policy":

add | **rm** | **set** | **show** | **stat** | **rename**

add cmp policy

Creates a classic or default syntax HTTP compression policy. When the policy matches an HTTP request or response, the action specified in the policy is performed on the transaction. The policy can be bound globally or to an entity. For the policy to have an effect, compression must be enabled on the service.

Synopsys

add cmp policy <name> -rule <expression> -resAction <string>

Arguments

name

Name of the HTTP compression policy. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

Can be changed after the policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my cmp policy" or 'my cmp policy').

rule

Expression that determines which HTTP requests or responses match the compression policy. Can be a classic expression or a default-syntax expression.

Note:

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

The following requirements apply only to the NetScaler CLI:

- * If the expression includes one or more spaces, enclose the entire expression in double quotation marks.
- * If the expression itself includes double quotation marks, escape the quotations by using the \\ character.
- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

resAction

The built-in or user-defined compression action to apply to the response when the policy matches a request or response.

Example

Example 1: `add cmp policy pdf_cmp -rule "RES.HTTP.HEADER Content-Type CONTAINS applicati`

rm cmp policy

Removes a user-defined HTTP compression policy.

Synopsys

rm cmp policy <name>

Arguments

name

Name of the HTTP compression policy to be removed.

Example

rm cmp policy cmp_policy_name The "show cmp policy" command shows all currently defined H

set cmp policy

Modifies the specified parameters of an HTTP compression policy. Note: Use the show cmp policy command to view all configured HTTP compression policies.

Synopsys

set cmp policy <name> [-rule <expression>] [-resAction <string>]

Arguments

name

Name of the HTTP compression policy to be modified.

rule

New rule to be associated with the HTTP compression policy. You can modify the existing rule or create a new rule.

resAction

The built-in or user-defined compression action to be associated with the policy.

Example

Example 1: add cmp policy pdf_cmp -rule "RES.HTTP.HEADER Content-Type CONTAINS applicati

show cmp policy

Displays details of all HTTP compression policies.

Synopsys

show cmp policy [<name>] show cmp policy stats - alias for 'stat cmp policy'

Arguments

name

Name of the HTTP compression policy for which to display details.

Outputs

stateflag

expressionType

Type of policy (Classic/Advanced)

rule

The request/response rule that will trigger the specified compression action.

reqAction

The compression action to be performed on requests.

resAction

The compression action to be performed on responses.

hits

Number of hits.

txbytes

Number of bytes transferred.

rxbytes

Number of bytes received.

clientTTLB

Total client TTLB value.

clientTransactions

Number of client transactions.

serverTTLB

Total server TTLB value.

serverTransactions

Number of server transactions.

piHits

Number of hits.

piTxBytes

Number of bytes transferred.

piRxBytes

Number of bytes received.

piCltTTLB

Total client TTLB value.

piCltTransactions

Number of client transactions.

piSvrTTLB

Total server TTLB value.

piSvrTransactions

Number of server transactions.

boundTo

The name of the entity to which the policy is bound.

activePolicy

priority

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

labelType

Type of policy label invocation.

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

description

Description of the policy

bindPolicyType

policyType

vserverType

builtin

Flag to determine if compression policy is builtin or not

isDefault

A value of true is returned if it is a default policy

devno

count

Example

```
> show cmp policy 4 Compression policies: 1) Name: ns_cmp_content_type
```

stat cmp policy

Displays compression statistics for all advanced compression policies, or for only the specified policy.

Synopsis

```
stat cmp policy [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full )]
```

Arguments

name

Name of the advanced compression policy for which to display statistics. If no name is specified, statistics for all advanced compression policies are shown.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Policy hits (Hits)

Number of hits on the policy

Policy undef hits (Undefhits)

Number of undef hits on the policy

Example

```
stat cmp policy
```

rename cmp policy

Renames a compression policy.

Synopsys

```
rename cmp policy <name>@ <newName>@
```

Arguments

name

Existing name of the policy.

newName

New name for the compression policy. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

Choose a name that reflects the function that the policy performs.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my cmp policy" or 'my cmp policy').

Example

```
rename cmp policy oldname newname
```

cmp policylabel

The following operations can be performed on "cmp policylabel":

add | **rm** | **bind** | **unbind** | **show** | **stat** | **rename**

add cmp policylabel

Creates a user-defined HTTP compression policy label for default-syntax policies. Policies that you bind to the label are evaluated only if you call the label from another policy.

Synopsys

```
add cmp policylabel <labelName> -type ( REQ | RES )
```

Arguments

labelName

Name of the HTTP compression policy label. Must begin with a letter, number, or the underscore character (_). Additional characters allowed, after the first character, are the hyphen (-), period (.) pound sign (#), space (), at sign (@), equals (=), and colon (:). The name must be unique within the list of policy labels for compression policies. Can be renamed after the policy label is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my cmp policylabel" or 'my cmp policylabel').

type

Type of packets (request packets or response) against which to match the policies bound to this policy label.

Possible values: REQ, RES

Example

```
add cmp policylabel cmp_pol_label -type REQ
```

rm cmp policylabel

Removes an HTTP compression policy label.

Synopsys

```
rm cmp policylabel <labelName>
```

Arguments

labelName

Name of the HTTP compression policy label to be removed.

Example

```
rm cmp policylabel cmp_pol_label
```

bind cmp policylabel

Binds a default-syntax HTTP compression policy to an HTTP compression policy label.

Synopsys

```
bind cmp policylabel <labelName> -policyName <string> -priority <positive_integer> [-gotoPriorityExpression <expression>] [-invoke (<labelType> <labelName>)]
```

Arguments

labelName

Name of the label to invoke if the current policy evaluates to TRUE.

policyName

Name of the compression policy to bind to the label.

priority

Integer specifying the priority of the policy. The lower the number, the higher the priority. By default, policies within a label are evaluated in the order of their priority numbers.

Minimum value: 1

Maximum value: 2147483647

gotoPriorityExpression

Expression or other value specifying the priority of the next policy, within the policy label, to evaluate if the current policy evaluates to TRUE. Specify one of the following values:

- * NEXT - Evaluate the policy with the next higher numbered priority.
- * END - Stop evaluation.
- * USE_INVOCATION_RESULT - Applicable if this policy invokes another policy label. If the final goto in the invoked policy label has a value of END, the evaluation stops. If the final goto is anything other than END, the current policy label performs a NEXT.
- * An expression that evaluates to a number.

If you specify an expression, its evaluation result determines the next policy to evaluate, as follows:

- * If the expression evaluates to a higher numbered priority, that policy is evaluated next.
- * If the expression evaluates to the priority of the current policy, the policy with the next higher priority number is evaluated next.
- * If the expression evaluates to a priority number that is numerically higher than the highest priority number, policy evaluation ends.

An UNDEF event is triggered if:

- * The expression is invalid.
- * The expression evaluates to a priority number that is numerically lower than the current policy's priority.
- * The expression evaluates to a priority number that is between the current policy's priority number (say, 30) and the highest priority number (say, 100), but does not match any configured priority number (for example, the expression evaluates to the number 85). This example assumes that the priority number increments by 10 for every successive policy, and therefore a priority number of 85 does not exist in the policy label.

invoke

Invoke policies bound to a virtual server or a user-defined policy label. After the invoked policies are evaluated, the flow returns to the policy with the next higher priority number in the original label.

labelType

Type of policy label invocation.

Possible values: reqvserver, resvserver, policylabel

Example

```
bind cmp policylabel cmp_pol_label -policyName cmp_pol -priority 1
```

unbind cmp policylabel

Unbinds a default-syntax HTTP compression policy from an HTTP compression policy label.

Synopsis

```
unbind cmp policylabel <labelName> <policyName> [-priority <positive_integer>]
```

Arguments

labelName

Name of the HTTP compression policy label from which to unbind the policy.

policyName

Name of the HTTP compression policy to unbind from the policy label.

priority

Priority of the NOPOLICY to unbind. Required only to unbind a NOPOLICY, if it has been bound to this policy label.

Minimum value: 1

Maximum value: 2147483647

Example

```
unbind cmp policylabel cmp_pol_label cmp_pol
```

show cmp policylabel

Displays details of configured HTTP compression policy labels.

Synopsis

```
show cmp policylabel [<labelName>]
```

Arguments

labelName

Name of the HTTP compression policy label for which to display details.

Outputs

stateflag**type**

Type of packets (request packets or response) against which to match the policies bound to this policy label.

numpol

Number of policies bound to label.

hits

Number of times policy label was invoked.

policyName

The compression policy name.

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

invoke

Invoke policies bound to a virtual server or a user-defined policy label. After the invoked policies are evaluated, the flow returns to the policy with the next higher priority number in the original label.

labelType

Type of policy label invocation.

labelName

Name of the label to invoke if the current policy evaluates to TRUE.

flowType

Flowtype of the bound compression policy.

description

Description of the policylabel

flags

devno

count

Example

```
i) show cmp policylabel cmp_pol_label ii) show cmp policylabel
```

stat cmp policylabel

Displays statistics for all compression policy labels.

Synopsys

```
stat cmp policylabel [<labelName>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

labelName

Name of the compression policy label for which to display statistics. If not specified, statistics are displayed for all compression policy labels.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Policy Label Hits (Hits)

Number of times policy label was invoked.

rename cmp policylabel

Renames a compression policylabel.

Synopsys

```
rename cmp policylabel <labelName>@ <newName>@
```

Arguments

labelName

Existing name of the policy label.

newName

New name for the compression policy label. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my cmp policylabel" or 'my cmp policylabel').

Example

```
rename cmp policylabel oldname newname
```

cmp stats

The following operations can be performed on "cmp stats":

show cmp stats

show cmp stats is an alias for stat cmp Displays compression statistics.

Synopsys

show cmp stats - alias for 'stat cmp'

Cache Redirection Commands

The entities on which you can perform NetScaler CLI operations:

- `cr policy`
- `cr vserver`

cr policy

The following operations can be performed on "cr policy":

add | **rm** | **set** | **show**

add cr policy

Creates a cache redirection policy. To associate the new policy with a cache redirection virtual server, use the bind cr vserver command.

Synopsis

```
add cr policy <policyName> -rule <expression>
```

Arguments

policyName

Name for the cache redirection policy. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at sign (@), equal sign (=), and hyphen (-) characters. Cannot be changed after the policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my policy" or 'my policy').

rule

Expression, or name of a named expression, against which traffic is evaluated. Written in the classic syntax.

Note:Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

The following requirements apply only to the NetScaler CLI:

- * If the expression includes one or more spaces, enclose the entire expression in double quotation marks.
- * If the expression itself includes double quotation marks, escape the quotations by using the \ character.
- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

rm cr policy

Removes a cache redirection policy. You can delete a user-defined cache redirection policy that is not bound to a cache redirection virtual server. If the policy is bound to a virtual server, you must first unbind the policy, and then remove it.

Synopsis

```
rm cr policy <policyName>
```

Arguments

policyName

Name of the cache redirection policy to remove.

set cr policy

Changes the specified parameters of an existing cache redirection policy.

Synopsys

set cr policy <policyName> -rule <expression>

Arguments

policyName

Name of the cache redirection policy to change.

rule

Expression, or name of a named expression, against which traffic is evaluated. Written in the classic syntax.

Note:

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator.

For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

The following requirements apply only to the NetScaler CLI:

- * If the expression includes one or more spaces, enclose the entire expression in double quotation marks.
- * If the expression itself includes double quotation marks, escape the quotations by using the \ character.
- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

show cr policy

Displays all existing cache redirection policies, or just the specified policy.

Synopsys

show cr policy [<policyName>]

Arguments

policyName

Name of the cache redirection policy to display. If this parameter is omitted, details of all the policies are displayed.

Outputs

rule

Expression, or name of a named expression, against which traffic is evaluated. Written in the classic syntax.

Note:Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

The following requirements apply only to the NetScaler CLI:

- * If the expression includes one or more spaces, enclose the entire expression in double quotation marks.
- * If the expression itself includes double quotation marks, escape the quotations by using the \ character.
- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

domain

Domain name.

vstype

Virtual server type.

csPolicyType

Indicates whether policy is PI or not.(used only during display)

builtin

devno

count

stateflag

cr vserver

The following operations can be performed on "cr vserver":

add | **rm** | **set** | **unset** | **bind** | **unbind** | **enable** | **disable** | **show** | **stat** | **rename**

add cr vserver

Creates a cache redirection virtual server.

Synopsys

```
add cr vserver <name> [-td <positive_integer>] <serviceType> [<IPAddress> <port> [-range <positive_integer>]] [-
cacheType <cacheType>] [-redirect <redirect>] [-onPolicyMatch ( CACHE | ORIGIN )] [-redirectURL <URL>] [-
cltTimeout <secs>] [-precedence ( RULE | URL )] [-arp ( ON | OFF )] [-map ( ON | OFF )] [-format ( ON | OFF )] [-via
( ON | OFF )] [-dnsVserverName <string>] [-destinationVServer <string>] [-domain <string>] [-soPersistenceTimeOut
<positive_integer>] [-soThreshold <positive_integer>] [-reuse ( ON | OFF )] [-state ( ENABLED | DISABLED )] [-
downStateFlush ( ENABLED | DISABLED )] [-backupVServer <string>] [-disablePrimaryOnDown ( ENABLED |
DISABLED )] [-l2Conn ( ON | OFF )] [-backendssl ( ENABLED | DISABLED )] [-Listenpolicy <expression>] [-
Listenpriority <positive_integer>] [-tcpProfileName <string>] [-httpProfileName <string>] [-comment <string>] [-
srcIPExpr <expression>] [-originUSIP ( ON | OFF )] [-usePortRange ( ON | OFF )] [-appflowLog ( ENABLED |
DISABLED )] [-netProfile <string>] [-icmpVsrResponse ( PASSIVE | ACTIVE )] [-RHlstate ( PASSIVE | ACTIVE )]
```

Arguments

name

Name for the cache redirection virtual server. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at sign (@), equal sign (=), and hyphen (-) characters. Can be changed after the cache redirection virtual server is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my server" or 'my server').

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

serviceType

Protocol (type of service) handled by the virtual server.

Possible values: HTTP, SSL, NNTP, HDX

IPAddress

IPv4 or IPv6 address of the cache redirection virtual server. Usually a public IP address. Clients send connection requests to this IP address.

Note: For a transparent cache redirection virtual server, use an asterisk (*) to specify a wildcard virtual server address.

port

Port number of the virtual server.

Default value: 80

Minimum value: 1

Maximum value: 65534

range

Number of consecutive IP addresses, starting with the address specified by the IPAddress parameter, to include in a range of addresses assigned to this virtual server.

Default value: 1

Minimum value: 1

Maximum value: 254

cacheType

Mode of operation for the cache redirection virtual server. Available settings function as follows:

* **TRANSPARENT** - Intercept all traffic flowing to the appliance and apply cache redirection policies to determine whether content should be served from the cache or from the origin server.

* **FORWARD** - Resolve the hostname of the incoming request, by using a DNS server, and forward requests for non-cacheable content to the resolved origin servers. Cacheable requests are sent to the configured cache servers.

* **REVERSE** - Configure reverse proxy caches for specific origin servers. Incoming traffic directed to the reverse proxy can either be served from a cache server or be sent to the origin server with or without modification to the URL.

Possible values: **TRANSPARENT**, **REVERSE**, **FORWARD**

Default value: **TRANSPARENT**

redirect

Type of cache server to which to redirect HTTP requests. Available settings function as follows:

* **CACHE** - Direct all requests to the cache.

* **POLICY** - Apply the cache redirection policy to determine whether the request should be directed to the cache or to the origin.

* **ORIGIN** - Direct all requests to the origin server.

Possible values: **CACHE**, **POLICY**, **ORIGIN**

Default value: **POLICY**

onPolicyMatch

Redirect requests that match the policy to either the cache or the origin server, as specified.

Note: For this option to work, you must set the cache redirection type to **POLICY**.

Possible values: **CACHE**, **ORIGIN**

Default value: **ORIGIN**

redirectURL

URL of the server to which to redirect traffic if the cache redirection virtual server configured on the NetScaler appliance becomes unavailable.

cltTimeout

Time-out value, in seconds, after which to terminate an idle client connection.

Maximum value: 31536000

precedence

Type of policy (URL or RULE) that takes precedence on the cache redirection virtual server. Applies only to cache redirection virtual servers that have both URL and RULE based policies. If you specify URL, URL based policies are applied first, in the following order:

1. Domain and exact URL
2. Domain, prefix and suffix
3. Domain and suffix
4. Domain and prefix
5. Domain only
6. Exact URL
7. Prefix and suffix
8. Suffix only
9. Prefix only
10. Default

If you specify RULE, the rule based policies are applied before URL based policies are applied.

Possible values: RULE, URL

Default value: RULE

arp

Use ARP to determine the destination MAC address.

Possible values: ON, OFF

map

Obsolete.

Possible values: ON, OFF

format

via

Insert a via header in each HTTP request. In the case of a cache miss, the request is redirected from the cache server to the origin server. This header indicates whether the request is being sent from a cache server.

Possible values: ON, OFF

Default value: ON

dnsVserverName

Name of the DNS virtual server that resolves domain names arriving at the forward proxy virtual server.

Note: This parameter applies only to forward proxy virtual servers, not reverse or transparent.

destinationVServer

Destination virtual server for a transparent or forward proxy cache redirection virtual server.

domain

Default domain for reverse proxies. Domains are configured to direct an incoming request from a specified source domain to a specified target domain. There can be several configured pairs of source and target domains. You can select one pair to be the default. If the host header or URL of an incoming request does not include a source domain, this option sends the request to the specified target domain.

soPersistenceTimeOut

Time-out, in minutes, for spillover persistence.

Minimum value: 2

Maximum value: 24

soThreshold

For CONNECTION (or) DYNAMICCONNECTION spillover, the number of connections above which the virtual server enters spillover mode. For BANDWIDTH spillover, the amount of incoming and outgoing traffic (in Kbps) before spillover. For HEALTH spillover, the percentage of active services (by weight) below which spillover occurs.

Minimum value: 1

reuse

Reuse TCP connections to the origin server across client connections. Do not set this parameter unless the Service Type parameter is set to HTTP. If you set this parameter to OFF, the possible settings of the Redirect parameter function as follows:

* CACHE - TCP connections to the cache servers are not reused.

* ORIGIN - TCP connections to the origin servers are not reused.

* POLICY - TCP connections to the origin servers are not reused.

If you set the Reuse parameter to ON, connections to origin servers and connections to cache servers are reused.

Possible values: ON, OFF

Default value: ON

state

Initial state of the cache redirection virtual server.

Possible values: ENABLED, DISABLED

Default value: ENABLED

downStateFlush

Perform delayed cleanup of connections to this virtual server.

Possible values: ENABLED, DISABLED

Default value: ENABLED

backupVServer

Name of the backup virtual server to which traffic is forwarded if the active server becomes unavailable.

disablePrimaryOnDown

Continue sending traffic to a backup virtual server even after the primary virtual server comes UP from the DOWN state.

Possible values: ENABLED, DISABLED

Default value: DISABLED

l2Conn

Use L2 parameters, such as MAC, VLAN, and channel to identify a connection.

Possible values: ON, OFF

backendssl

Decides whether the backend connection made by NS to the origin server will be HTTP or SSL. Applicable only for SSL type CR Forward proxy vserver.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Listenpolicy

String specifying the listen policy for the cache redirection virtual server. Can be either an in-line expression or the name of a named expression.

Default value: "none"

Listenpriority

Priority of the listen policy specified by the Listen Policy parameter. The lower the number, higher the priority.

Default value: 101

Minimum value: 0

Maximum value: 100

tcpProfileName

Name of the profile containing TCP configuration information for the cache redirection virtual server.

httpProfileName

Name of the profile containing HTTP configuration information for cache redirection virtual server.

comment

Comments associated with this virtual server.

srcIPExpr

Expression used to extract the source IP addresses from the requests originating from the cache. Can be either an in-line expression or the name of a named expression.

originUSIP

Use the client's IP address as the source IP address in requests sent to the origin server.

Note: You can enable this parameter to implement fully transparent CR deployment.

Possible values: ON, OFF

Default value: OFF

usePortRange

Use a port number from the port range (set by using the set ns param command, or in the Create Virtual Server (Cache Redirection) dialog box) as the source port in the requests sent to the origin server.

Possible values: ON, OFF

Default value: OFF

appflowLog

Enable logging of AppFlow information.

Possible values: ENABLED, DISABLED

Default value: ENABLED

netProfile

Name of the network profile containing network configurations for the cache redirection virtual server.

icmpVsrResponse

Criterion for responding to PING requests sent to this virtual server. If ACTIVE, respond only if the virtual server is available. If PASSIVE, respond even if the virtual server is not available.

Possible values: PASSIVE, ACTIVE

Default value: PASSIVE

RHlstate

A host route is injected according to the setting on the virtual servers

* If set to PASSIVE on all the virtual servers that share the IP address, the appliance always injects the hostroute.

* If set to ACTIVE on all the virtual servers that share the IP address, the appliance injects even if one virtual server is UP.

* If set to ACTIVE on some virtual servers and PASSIVE on the others, the appliance, injects even if one virtual server set to ACTIVE is UP.

Possible values: PASSIVE, ACTIVE

Default value: PASSIVE

rm cr vsrver

Removes a virtual server.

Synopsys

```
rm cr vsrver <name>@ ...
```

Arguments

name

Name of the virtual server to be removed.

Example

```
rm vsrver cr_vip
```

set cr vsrver

Changes the specified settings of the cache redirection virtual server.

Synopsys

```
set cr vsrver <name> [-IPAddress <ip_addr|ipv6_addr|*>] [-redirect <redirect>] [-onPolicyMatch ( CACHE | ORIGIN )] [-precedence ( RULE | URL )] [-arp ( ON | OFF )] [-via ( ON | OFF )] [-dnsVserverName <string>] [-destinationVServer <string>] [-domain <string>] [-reuse ( ON | OFF )] [-backupVServer <string>] [-disablePrimaryOnDown ( ENABLED | DISABLED )] [-redirectURL <URL>] [-cltTimeout <secs>] [-downStateFlush ( ENABLED | DISABLED )] [-l2Conn ( ON | OFF )] [-backendssl ( ENABLED | DISABLED )] [-Listenpolicy <expression>] [-Listenpriority <positive_integer>] [-tcpProfileName <string>] [-httpProfileName <string>] [-netProfile <string>] [-comment <string>] [-srcIPExpr <expression>] [-originUSIP ( ON | OFF )] [-usePortRange ( ON | OFF )] [-appflowLog ( ENABLED | DISABLED )] [-icmpVsrResponse ( PASSIVE | ACTIVE )] [-RHlstate ( PASSIVE | ACTIVE )]
```

Arguments

name

Name of the cache redirection virtual server.

IPAddress

New IPv4 or IPv6 address of the cache redirection virtual server. Usually a public IP address. Clients send connection requests to this IP address.

redirect

Type of server to which to redirect HTTP requests. Available settings function as follows: * CACHE - Direct all requests to the cache.* POLICY - Apply the cache redirection policy to determine whether the request should be directed to the cache or to the origin.* ORIGIN - Direct all requests to the origin server.

Possible values: CACHE, POLICY, ORIGIN

Default value: POLICY

onPolicyMatch

Redirect requests that match the policy to either the cache or the origin server, as specified.

Note: For this option to work, you must set the cache redirection type to POLICY.

Possible values: CACHE, ORIGIN

Default value: ORIGIN

precedence

Type of policy (URL or RULE) that takes precedence on the cache redirection virtual server. You can use this argument only when configuring cache redirection on the specified virtual server. It applies only if both URL and RULE based policies have been configured on the same virtual server. Available settings function as follows:URL - The incoming request is matched against the URL-based policies before it is matched against the rule-based policies.

For URL based policies, the precedence hierarchy is:

1. Domain and exact URL
2. Domain, prefix and suffix
3. Domain and suffix
4. Domain and prefix
5. Domain only
6. Exact URL
7. Prefix and suffix
8. Suffix only
9. Prefix only
10. Default

RULE - The incoming request is matched against the rule-based policies before it is matched against the URL-based policies.

Possible values: RULE, URL

Default value: RULE

arp

Use ARP to determine the destination MAC address. Specify OFF to use the incoming destination MAC address, or ON to use ARP to determine the destination MAC address.

Possible values: ON, OFF

via

Insert a via header in each HTTP request. In the case of a cache miss, the request is redirected from the cache server to the origin server. This header indicates whether the request is being sent from a cache server.

Possible values: ON, OFF

Default value: ON

dnsVserverName

Name of the DNS virtual server that resolves domain names arriving at the forward proxy virtual server.

Note: This parameter applies only to forward proxy virtual servers, not reverse or transparent.

destinationVServer

Destination virtual server for a transparent or forward proxy cache redirection virtual server.

domain

Default domain for reverse proxies. Domains are configured to direct incoming requests from a specified source domain to a specified target domain. There can be several configured pairs of source and target domains. You can select one pair to be the default. If the host header or URL of an incoming request does not include a source domain, this option sends the request to the specified target domain.

reuse

Reuse TCP connections to the origin server across client connections

Possible values: ON, OFF

Default value: ON

backupVServer

Name of the backup virtual server to which traffic is forwarded if the active server becomes unavailable.

disablePrimaryOnDown

Continue sending traffic to a backup virtual server even after the primary virtual server comes UP from the DOWN state.

Possible values: ENABLED, DISABLED

Default value: DISABLED

redirectURL

URL of the server to which to redirect traffic if the cache redirection virtual server in the NetScaler becomes unavailable.

cltTimeout

Time-out value, in seconds, after which an idle client connection is terminated.

Maximum value: 31536000

downStateFlush

Perform delayed cleanup of connections to this virtual server.

Possible values: ENABLED, DISABLED

Default value: ENABLED

l2Conn

Use L2 parameters, such as MAC, VLAN, and channel to identify a connection.

Possible values: ON, OFF

backendssl

Decides whether the backend connection made by NS to the origin server will be HTTP or SSL. Applicable only for SSL type CR Forward proxy vserver.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Listenpolicy

String specifying the listen policy for the cache redirection virtual server. Can be either an in-line expression or the name of a named expression.

Default value: "none"

Listenpriority

Priority of the listen policy specified by the Listen Policy parameter. The lower the number, higher the priority.

Default value: 101

Minimum value: 0

Maximum value: 100

tcpProfileName

Name of the profile containing TCP configuration information for the cache redirection virtual server.

httpProfileName

Name of the profile containing HTTP configuration information for cache redirection virtual server.

netProfile

Name of the network profile containing network configurations for the cache redirection virtual server.

comment

Comments associated with this virtual server.

srcIPExpr

Expression used to extract the source IP addresses from the requests originating from the cache. Can be either an in-line expression or the name of a named expression.

originUSIP

Use the client's IP address as the source IP address in requests sent to the origin server.

Note: You can enable this parameter to implement fully transparent CR deployment.

Possible values: ON, OFF

Default value: OFF

usePortRange

Use a port number from the port range (set by using the set ns param command, or in the Create Virtual Server (Cache Redirection) dialog box) as the source port in the requests sent to the origin server.

Possible values: ON, OFF

Default value: OFF

appflowLog

Enable logging of AppFlow information.

Possible values: ENABLED, DISABLED

Default value: ENABLED

icmpVsrResponse

Criterion for responding to PING requests sent to this virtual server. If ACTIVE, respond only if the virtual server is available. If PASSIVE, respond even if the virtual server is not available.

Possible values: PASSIVE, ACTIVE

Default value: PASSIVE

RHlstate

A host route is injected according to the setting on the virtual servers

* If set to PASSIVE on all the virtual servers that share the IP address, the appliance always injects the hostroute.

* If set to ACTIVE on all the virtual servers that share the IP address, the appliance injects even if one virtual server is UP.

* If set to ACTIVE on some virtual servers and PASSIVE on the others, the appliance, injects even if one virtual server set to ACTIVE is UP.

Possible values: PASSIVE, ACTIVE

Default value: PASSIVE

unset cr vserver

Restores the specified parameters of a cache redirection virtual server to their default values. To unset all except the Name parameter, do not specify a value for any other parameter. Refer to the set cr vserver command for a description of the parameters..Refer to the set cr vserver command for meanings of the arguments.

Synopsis

```
unset cr vserver <name> [-dnsVserverName] [-destinationVServer] [-domain] [-backupVServer] [-cltTimeout] [-redirectURL] [-l2Conn] [-backendssl] [-originUSIP] [-usePortRange] [-srcIPExpr] [-tcpProfileName] [-httpProfileName] [-appflowLog] [-netProfile] [-icmpVsrResponse] [-redirect] [-onPolicyMatch] [-precedence] [-arp] [-via] [-reuse] [-disablePrimaryOnDown] [-downStateFlush] [-Listenpolicy] [-Listenpriority] [-comment] [-RHlstate]
```

bind cr vserver

Binds a cache redirection policy to a cache redirection virtual server.

Synopsis

```
bind cr vserver <name> [-lbvserver <string> | (-policyName <string> [-priority <positive_integer>])] | <targetVserver>]
```

Arguments

name

Name of the cache redirection virtual server to which to bind the cache redirection policy.

lbvserver

Name of the virtual server to which content is forwarded. Applicable only if the policy is a map policy and the cache redirection virtual server is of type REVERSE.

policyName

Name of the cache redirection policy that you are binding.

targetVserver

Name of the virtual server to which content is forwarded. Applicable only if the policy is a map policy and the cache redirection virtual server is of type REVERSE.

priority

An unsigned integer that determines the priority of the policy relative to other policies bound to this cache redirection virtual server. The lower the value, higher the priority. Note: This option is available only when binding content switching, filtering, and compression policies to a cache redirection virtual server.

Minimum value: 0

unbind cr vserver

Unbinds a cache redirection policy from a cache redirection virtual server.

Synopsis

```
unbind cr vserver <name> [-policyName <string> | -lbvserver <string>]
```

Arguments

name

Name of the cache redirection virtual server from which to unbind the policy.

policyName

Name of the cache redirection policy that you are unbinding.

lbvserver

The virtual server name (created with the add lb vserver command) to which content will be switched.

Default value: "default_lb"

enable cr vserver

Enables a cache redirection virtual server. Note: Virtual servers, when added, are enabled by default.

Synopsis

```
enable cr vserver <name>@
```

Arguments

name

Name of the cache redirection virtual server to be enabled.

Example

```
enable vserver cr_vip
```

disable cr vserver

Disables a cache redirection virtual server.

Synopsis

```
disable cr vserver <name>@
```

Arguments

name

Name of the cache redirection virtual server to be disabled. (Because the virtual server is still configured, you can reenale it.)

Note: The appliance still responds to ARP and ping requests sent to the IP address of this virtual server.

Example

```
disable vserver cr_vip
```

show cr vserver

Displays cache redirection virtual server information. To display information about all configured cache redirection virtual servers, do not include a parameter. To display detailed information about a specific virtual server, use the name parameter to specify the name of the virtual server.

Synopsys

```
show cr vserver [<name>]
```

Arguments

name

Name of a cache redirection virtual server about which to display detailed information.

Outputs

IPAddress

The IP address of the virtual server.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

stateflag

value

The ssl card status for the transparent ssl cr vserver.

port

Port number of the virtual server.

range

Number of consecutive IP addresses, starting with the address specified by the IP Address parameter, to include in a range of addresses assigned to this virtual server.

serviceType

Protocol (type of service) handled by the virtual server.

ngname

Nodegroup devno to which this crvserver belongs to

type

Virtual server type.

vsvrcfgflags

Contains the config info of vserver to be used at validation

state

Initial state of the cache redirection virtual server.

status

Status.

cacheType

Mode of operation for the cache redirection virtual server. Available settings function as follows:

- * TRANSPARENT - Intercept all traffic flowing to the appliance and apply cache redirection policies to determine whether content should be served from the cache or from the origin server.
- * FORWARD - Resolve the hostname of the incoming request, by using a DNS server, and forward requests for non-cacheable content to the resolved origin servers. Cacheable requests are sent to the configured cache servers.
- * REVERSE - Configure reverse proxy caches for specific origin servers. Incoming traffic directed to the reverse proxy can either be served from a cache server or be sent to the origin server with or without modification to the URL.

redirect

Type of cache server to which to redirect HTTP requests. Available settings function as follows:

- * CACHE - Direct all requests to the cache.
- * POLICY - Apply the cache redirection policy to determine whether the request should be directed to the cache or to the origin.
- * ORIGIN - Direct all requests to the origin server.

onPolicyMatch

Redirect requests that match the policy to either the cache or the origin server, as specified.

Note: For this option to work, you must set the cache redirection type to POLICY.

precedence

Type of policy (URL or RULE) that takes precedence on the cache redirection virtual server. Applies only to cache redirection virtual servers that have both URL and RULE based policies. If you specify URL, URL based policies are applied first, in the following order:

1. Domain and exact URL
2. Domain, prefix and suffix
3. Domain and suffix
4. Domain and prefix
5. Domain only
6. Exact URL
7. Prefix and suffix
8. Suffix only
9. Prefix only
10. Default

If you specify RULE, the rule based policies are applied before URL based policies are applied.

redirectURL

URL of the server to which to redirect traffic if the cache redirection virtual server configured on the NetScaler appliance becomes unavailable.

authentication

Authentication.

homePage

Home page.

dnsVserverName

Name of the DNS virtual server that resolves domain names arriving at the forward proxy virtual server.

Note: This parameter applies only to forward proxy virtual servers, not reverse or transparent.

domain

Default domain for reverse proxies. Domains are configured to direct an incoming request from a specified source domain to a specified target domain. There can be several configured pairs of source and target domains. You can select one pair to be the default. If the host header or URL of an incoming request does not include a source domain, this option sends the request to the specified target domain.

rule

Rule.

policyName

Policies bound to this vserver.

hits

Number of hits.

serviceName

Service name.

weight

Weight for this service.

cacheVserver

Name of the default cache virtual server to which to redirect requests (the default target of the cache redirection virtual server).

targetVserver

The CSW target server names.

backupVServer

Name of the backup virtual server to which traffic is forwarded if the active server becomes unavailable.

priority

The priority for the policy.

cltTimeout

Time-out value, in seconds, after which to terminate an idle client connection.

soMethod

The spillover factor. When the main virtual server reaches this spillover threshold, it will give further traffic to the backupvserver.

soPersistence

The state of spillover persistence.

soPersistenceTimeout

The spillover persistence entry timeout.

soThreshold

The spillover threshold value.

reuse

Reuse TCP connections to the origin server across client connections. Do not set this parameter unless the Service Type parameter is set to HTTP. If you set this parameter to OFF, the possible settings of the Redirect parameter function as follows:

- * CACHE - TCP connections to the cache servers are not reused.
- * ORIGIN - TCP connections to the origin servers are not reused.
- * POLICY - TCP connections to the origin servers are not reused.

If you set the Reuse parameter to ON, connections to origin servers and connections to cache servers are reused.

arp**destinationVServer**

Destination virtual server for a transparent or forward proxy cache redirection virtual server.

via

Insert a via header in each HTTP request. In the case of a cache miss, the request is redirected from the cache server to the origin server. This header indicates whether the request is being sent from a cache server.

downStateFlush

Perform delayed clean up of connections on this vserver.

disablePrimaryOnDown

Tells whether traffic will continue reaching backup vservers even after primary comes UP from DOWN state.

l2Conn

Use L2 parameters, such as MAC, VLAN, and channel to identify a connection.

backendssl

Decides whether the backend connection made by NS to the origin server will be HTTP or SSL. Applicable only for SSL type CR Forward proxy vserver.

comment

Comments associated with this virtual server.

Listenpolicy

The string is listenpolicy configured for CR vserver

Listenpriority

This parameter is the priority for listen policy of CR Vserver.

tcpProfileName

Name of the profile containing TCP configuration information for the cache redirection virtual server.

httpProfileName

Name of the profile containing HTTP configuration information for cache redirection virtual server.

srcIPExpr

Expression used to extract the source IP addresses from the requests originating from the cache. Can be either an in-line expression or the name of a named expression.

originUSIP

Use the client's IP address as the source IP address in requests sent to the origin server.

Note: You can enable this parameter to implement fully transparent CR deployment.

usePortRange

Use a port number from the port range (set by using the set ns param command, or in the Create Virtual Server (Cache Redirection) dialog box) as the source port in the requests sent to the origin server.

appflowLog

Enable logging of AppFlow information.

netProfile

Name of the network profile containing network configurations for the cache redirection virtual server.

icmpVsrResponse

Criterion for responding to PING requests sent to this virtual server. If ACTIVE, respond only if the virtual server is available. If PASSIVE, respond even if the virtual server is not available.

RHIstate

A host route is injected according to the setting on the virtual servers

* If set to PASSIVE on all the virtual servers that share the IP address, the appliance always injects the host route.

* If set to ACTIVE on all the virtual servers that share the IP address, the appliance injects even if one virtual server is UP.

* If set to ACTIVE on some virtual servers and PASSIVE on the others, the appliance, injects even if one virtual server set to ACTIVE is UP.

lbvserver

The Default target server name.

inherited

On State describes that policy bound is inherited from global binding.

devno**count****stat cr vserver**

Displays statistics for all cache redirection virtual servers or for the cache redirection virtual server specified by the name parameter.

Synopsis

```
stat cr vserver [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full )]
```

Arguments

name

Name of a specific cache redirection virtual server.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

IP address (IP)

The IP address on which the service is running.

Port (port)

The port on which the service is running.

Vserver protocol (Protocol)

Protocol associated with the vservers

State

Current state of the server. Possible values are UP, DOWN, UNKNOWN, OFS(Out of Service), TROFS (Transition Out of Service), TROFS_DOWN(Down When going Out of Service)

Requests (Req)

Total number of requests received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

Responses (Rsp)

Number of responses received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

Request bytes (Reqb)

Total number of request bytes received on this service or virtual server.

Response bytes (Rspb)

Number of response bytes received by this service or virtual server.

rename cr vserver

Renames a cache redirection virtual server.

Synopsys

```
rename cr vserver <name>@ <newName>@
```

Arguments

name

Existing name of the cache redirection virtual server.

newName

New name for the cache redirection virtual server. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at sign (@), equal sign (=), and hyphen (-) characters. If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my name? or ?my name?).

Example

```
rename cr vserver vscr1 vscrnew
```


Content Switching Commands

The entities on which you can perform NetScaler CLI operations:

- o `cs action`
- o `cs parameter`
- o `cs policy`
- o `cs policylabel`
- o `cs vserver`

cs action

The following operations can be performed on "cs action":

add | **rm** | **set** | **unset** | **show** | **rename**

add cs action

Creates an action that indicates the target load balancing virtual server. This action is used to specify the target load balancing virtual server while defining a policy to support multiple policy bind support.

Synopsys

`add cs action <name> (-targetLBVserver <string> | -targetVserverExpr <expression>) [-comment <string>]`

Arguments

name

Name for the content switching action. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at sign (@), equal sign (=), and hyphen (-) characters. Can be changed after the content switching action is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my action" or 'my action').

targetLBVserver

Name of the load balancing virtual server to which the content is switched.

targetVserverExpr

Information about this content switching action.

comment

Comments associated with this cs action.

Example

```
add cs action -targetLBVserver act1 lb1
```

rm cs action

Removes a content switching action.

Synopsys

`rm cs action <name>`

Arguments

name

Name of the cs action.

Example

```
rm cs action act_before
```

set cs action

Modifies the configuration settings of a content switching action.

Synopsis

```
set cs action <name> (-targetLBVserver <string> | -targetVserverExpr <expression>) [-comment <string>]
```

Arguments

name

Name for the content switching action. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at sign (@), equal sign (=), and hyphen (-) characters. Can be changed after the content switching action is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my action? or ?my action?).

targetLBVserver

Name of the load balancing virtual server to which the content is switched.

targetVserverExpr

Information about this content switching action.

comment

Comments associated with this cs action.

Example

```
set cs action act1 -targetLBVserver lb2 -comment 'for url'
```

unset cs action

Use this command to remove cs action settings. Refer to the set cs action command for meanings of the arguments.

Synopsis

```
unset cs action <name> -comment
```

show cs action

Displays the configuration settings of the specified content switching action or lists all the content switching actions configured on the appliance.

Synopsis

```
show cs action [<name>]
```

Arguments

name

Name of the content switching action.

Outputs

stateflag

targetLBVserver

Target LB vserver name.

targetVserverExpr

Target LB vserver expression.

hits

The number of times the action has been taken.

referenceCount

The number of references to the action.

undefHits

The number of times the action resulted in UNDEF.

builtin**comment**

Comments associated with this cs action.

devno**count**

Example

```
show cs action
```

rename cs action

Renames a content switching action.

Synopsis

```
rename cs action <name>@ <newName>@
```

Arguments

name

Existing name of the content switching action.

newName

New name for the content switching action. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at sign (@), equal sign (=), and hyphen (-) characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my name? or ?my name?).

Example

```
rename cs action oldname newname
```

cs parameter

The following operations can be performed on "cs parameter":

[set](#) | [unset](#) | [show](#)

set cs parameter

Sets the status of the state update parameter for the server. By default, the content switching virtual server is always UP, regardless of the state of the load balancing virtual servers bound to it. This command enables the virtual server to check the status of the attached load balancing server for state information.

Synopsys

```
set cs parameter -stateupdate ( ENABLED | DISABLED )
```

Arguments

stateupdate

Specifies whether the virtual server checks the attached load balancing server for state information.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
set cs parameter -stateupdate (ENABLED|DISABLED)
```

unset cs parameter

Use this command to remove cs parameter settings. Refer to the set cs parameter command for meanings of the arguments.

Synopsys

```
unset cs parameter -stateupdate
```

show cs parameter

Show CS parameters

Synopsys

```
show cs parameter
```

Outputs

stateupdate

Specifies whether the virtual server checks the attached load balancing server for state information.

Example

```
show cs parameter
```

cs policy

The following operations can be performed on "cs policy":

add | **rm** | **set** | **unset** | **show** | **rename**

add cs policy

Creates a new content switching policy. You use this policy to manage content switching on a virtual server.

Synopsis

```
add cs policy <policyName> [-url <string> | -rule <expression> | -action <string>] [-domain <string>] [-logAction <string>]
```

Arguments

policyName

Name for the content switching policy. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at sign (@), equal sign (=), and hyphen (-) characters. Cannot be changed after a policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my policy? or ?my policy?).

url

URL string that is matched with the URL of a request. Can contain a wildcard character. Specify the string value in the following format: [[prefix] [*]] [.suffix].

rule

Expression, or name of a named expression, against which traffic is evaluated. Written in the classic or default syntax.

Note:

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

The following requirements apply only to the NetScaler CLI:

- * If the expression includes one or more spaces, enclose the entire expression in double quotation marks.
- * If the expression itself includes double quotation marks, escape the quotations by using the \ character.
- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

domain

The domain name. The string value can range to 63 characters.

action

Content switching action that names the target load balancing virtual server to which the traffic is switched.

logAction

The log action associated with the content switching policy

Example

To match the requests that have URL "/", you would enter the following command: `add cs po`

rm cs policy

Removes a content switching policy. You can delete a user-defined content switching policy that is not bound to a content switching virtual server. If the policy is bound to a virtual server, you must first unbind the policy, and then remove it.

Synopsis

```
rm cs policy <policyName>
```

Arguments

policyName

Name of the content switching policy to be removed.

set cs policy

Changes an existing content switching policy.

Synopsis

```
set cs policy <policyName> [-url <string> | -rule <expression>] [-domain <string>] [-action <string>] [-logAction <string>]
```

Arguments

policyName

Name of the content switching policy.

url

The URL, with wildcards.

rule

The condition for applying this policy.

domain

The domain name.

action

The content switching action name.

logAction

The log action associated with the content switching policy

unset cs policy

Unset logaction for existing content switching policy..Refer to the set cs policy command for meanings of the arguments.

Synopsis

```
unset cs policy <policyName> [-logAction] [-url] [-rule] [-domain] [-action]
```

Example

```
unset cs policy pol9 -logAction
```

show cs policy

Displays all existing content switching policies, or just the specified policy.

Synopsys

show cs policy [<policyName>]

Arguments

policyName

Name of the content switching policy to display. If this parameter is omitted, details of all the policies are displayed.

Outputs

url

The URL with wildcards.

rule

The condition for applying this policy.

domain

The domain name.

action

The CS action name.

vstype

Virtual server type.

hits

Total number of hits.

piHits

Total number of hits.

bindHits

Total number of hits.

piPolicyhits

bind hits for PI CS Policy.

labelName

Name of the label invoked.

labelType

The invocation type.

target

Target flag

priority

priority of bound policy

flag

stateflag

activePolicy

Indicates whether policy is bound or not.

csPolicyType

Indicates whether policy is PI or not.(used only during display)

logAction

The log action associated with the content switching policy

devno

count

rename cs policy

Rename a content switching policy.

Synopsys

```
rename cs policy <policyName>@ <newName>@
```

Arguments

policyName

The name of the content switching policy.

newName

The new name of the content switching policy.

Example

```
rename cs policy oldname newname
```

cs policylabel

The following operations can be performed on "cs policylabel":

add | **rm** | **bind** | **unbind** | **show** | **rename**

add cs policylabel

Adds a content switching policy label.

Synopsis

add cs policylabel <labelName> <cspolicylabeltype>

Arguments

labelName

Name for the policy label. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at sign (@), equal sign (=), and hyphen (-) characters.

The label name must be unique within the list of policy labels for content switching.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, \\?my label\\? or \\?my policylabel\\?).

cspolicylabeltype

Protocol supported by the policy label. All policies bound to the policy label must either match the specified protocol or be a subtype of that protocol. Available settings function as follows:

- * HTTP - Supports policies that process HTTP traffic. Used to access unencrypted Web sites. (The default.)
- * SSL - Supports policies that process HTTPS/SSL encrypted traffic. Used to access encrypted Web sites.
- * TCP - Supports policies that process any type of TCP traffic, including HTTP.
- * SSL_TCP - Supports policies that process SSL-encrypted TCP traffic, including SSL.
- * UDP - Supports policies that process any type of UDP-based traffic, including DNS.
- * DNS - Supports policies that process DNS traffic.
- * ANY - Supports all types of policies except HTTP, SSL, and TCP.
- * SIP_UDP - Supports policies that process UDP based Session Initiation Protocol (SIP) traffic. SIP initiates, manages, and terminates multimedia communications sessions, and has emerged as the standard for Internet telephony (VoIP).
- * RTSP - Supports policies that process Real Time Streaming Protocol (RTSP) traffic. RTSP provides delivery of multimedia and other streaming data, such as audio, video, and other types of streamed media.
- * RADIUS - Supports policies that process Remote Authentication Dial In User Service (RADIUS) traffic. RADIUS supports combined authentication, authorization, and auditing services for network management.
- * MYSQL - Supports policies that process MYSQL traffic.
- * MSSQL - Supports policies that process Microsoft SQL traffic.

Possible values: HTTP, TCP, RTSP, SSL, SSL_TCP, UDP, DNS, SIP_UDP, ANY, RADIUS, RDP, MYSQL, MSSQL, ORACLE, DIAMETER, SSL_DIAMETER, FTP, DNS_TCP

Example

```
add cs policylabel trans_http_url HTTP
```

rm cs policylabel

Removes a content switching policy label.

Synopsis

```
rm cs policylabel <labelName>
```

Arguments

labelName

Name of the label to be removed.

Example

```
rm cs policylabel trans_http_url
```

bind cs policylabel

Binds a content switching policy to a content switching policy label.

Synopsis

```
bind cs policylabel <labelName> <policyName> <priority> [-targetVserver <string> | (-invoke (<labelType>  
<labelName>)) ] [-gotoPriorityExpression <expression>]
```

Arguments

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

policyName

Name of the content switching policy to bind to the content switching policy label.

priority

Unsigned integer that determines the priority of the policy relative to other policies in this policy label. Smaller the number, higher the priority.

Minimum value: 1

Maximum value: 2147483647

targetVserver

Name of the virtual server to which to forward requests that match the policy.

gotoPriorityExpression

Expression or other value specifying the priority of the next policy to be evaluated if the current policy rule evaluates to TRUE. Alternatively, you can specify one of the following values:

* NEXT - Go to the policy with the next higher priority.

* END - End evaluation. (This is the default. Evaluation stops if the gotoPriorityExpression parameter is not set.)

* USE_INVOCATION_RESULT - Applicable if this entry invokes another policy label. If the final goto in the invoked policy label has a value of END, evaluation stops. If the final goto is anything other than END, the current policy label performs a NEXT.

If you specify an expression, its result must be a number. In that case, the next action is determined as follows:

- * If the expression evaluates to the priority of a policy with a lower priority (larger priority number) than the current policy, that policy is evaluated next.
- * If the expression evaluates to a priority of the current policy, policy with the next highest priority is evaluated.

An UNDEF event is triggered if:

- * The expression cannot be evaluated.
- * The expression evaluates to a number that is smaller than the highest priority in the policy bank but is not same as any policy's priority.
- * The expression evaluates to a number that is smaller than the current policy's priority.

invoke

Invoke other policy labels. After evaluating the policies in the invoked policy label, the appliance continues to evaluate policies that are bound to the current policy label (the selected bind point).

labelType

Type of policy label invocation.

Possible values: policylabel

Example

```
i)      bind cs policylabel cs_lab lbvs_1 pol_cs 1 2
```

unbind cs policylabel

Unbinds a content switching policy from a content switching policy label.

Synopsys

```
unbind cs policylabel <labelName> <policyName>
```

Arguments

labelName

Name of the policy label from which to unbind a content switching policy.

policyName

Name of the content switching policy to unbind from the label.

Example

```
unbind cs policylabel cs_lab pol_cs
```

show cs policylabel

Displays all the content switching policy labels, or just the specified policy label.

Synopsys

```
show cs policylabel [<labelName>]
```

Arguments

labelName

Name of the content switching policy label to display.

Outputs

cspolicylabeltype

Protocol supported by the policy label. All policies bound to the policy label must either match the specified protocol or be a subtype of that protocol. Available settings function as follows:

- * HTTP - Supports policies that process HTTP traffic. Used to access unencrypted Web sites. (The default.)
- * SSL - Supports policies that process HTTPS/SSL encrypted traffic. Used to access encrypted Web sites.
- * TCP - Supports policies that process any type of TCP traffic, including HTTP.
- * SSL_TCP - Supports policies that process SSL-encrypted TCP traffic, including SSL.
- * UDP - Supports policies that process any type of UDP-based traffic, including DNS.
- * DNS - Supports policies that process DNS traffic.
- * ANY - Supports all types of policies except HTTP, SSL, and TCP.
- * SIP_UDP - Supports policies that process UDP based Session Initiation Protocol (SIP) traffic. SIP initiates, manages, and terminates multimedia communications sessions, and has emerged as the standard for Internet telephony (VoIP).
- * RTSP - Supports policies that process Real Time Streaming Protocol (RTSP) traffic. RTSP provides delivery of multimedia and other streaming data, such as audio, video, and other types of streamed media.
- * RADIUS - Supports policies that process Remote Authentication Dial In User Service (RADIUS) traffic. RADIUS supports combined authentication, authorization, and auditing services for network management.
- * MYSQL - Supports policies that process MYSQL traffic.
- * MSSQL - Supports policies that process Microsoft SQL traffic.

stateflag

numpol

number of polices bound to label.

hits

Number of times policy label was invoked.

policyName

Name of the content switching policy.

priority

Specifies the priority of the policy.

targetVserver

Name of the virtual server to which to forward requests that match the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

labelType

Type of policy label invocation.

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

invoke

devno

count

Example

```
i)      show cs policylabel cs_lab      ii)      show cs policylabel
```

rename cs policylabel

Rename a content switching policy label.

Synopsys

```
rename cs policylabel <labelName>@ <newName>@
```

Arguments

labelName

The name of the content switching policylabel.

newName

The new name of the content switching policylabel.

Example

```
rename cs policylabel oldname newname
```

cs vserver

The following operations can be performed on "cs vserver":

add | **rm** | **set** | **unset** | **bind** | **unbind** | **enable** | **disable** | **show** | **stat** | **rename**

add cs vserver

Creates a content switching virtual server.

Synopsys

```
add cs vserver <name> [-td <positive_integer>] <serviceType> ((<IPAddress> [-range <positive_integer>]) | (-
IPPattern <ippat> -IPMask <ipmask>)) <port> [-state ( ENABLED | DISABLED )] [-stateupdate ( ENABLED |
DISABLED )] [-cacheable ( YES | NO )] [-redirectURL <URL>] [-cltTimeout <secs>] [-precedence ( RULE | URL )] [-
caseSensitive ( ON | OFF )] [-soMethod <soMethod>] [-soPersistence ( ENABLED | DISABLED )] [-
soPersistenceTimeout <positive_integer>] [-soThreshold <positive_integer>] [-soBackupAction <soBackupAction>] [-
redirectPortRewrite ( ENABLED | DISABLED )] [-downStateFlush ( ENABLED | DISABLED )] [-backupVServer
<string>] [-disablePrimaryOnDown ( ENABLED | DISABLED )] [-insertVserverIPPort <insertVserverIPPort>
[<vipHeader>] ] [-rtspNat ( ON | OFF )] [-AuthenticationHost <string>] [-Authentication ( ON | OFF )] [-Listenpolicy
<expression>] [-Listenpriority <positive_integer>] [-authn401 ( ON | OFF )] [-authnVsName <string>] [-push (
ENABLED | DISABLED )] [-pushVserver <string>] [-pushLabel <expression>] [-pushMultiClients ( YES | NO )] [-
tcpProfileName <string>] [-httpProfileName <string>] [-dbProfileName <string>] [-oracleServerVersion ( 10G | 11G )]
[-comment <string>] [-mssqlServerVersion <mssqlServerVersion>] [-I2Conn ( ON | OFF )] [-mysqlProtocolVersion
<positive_integer>] [-mysqlServerVersion <string>] [-mysqlCharacterSet <positive_integer>] [-
mysqlServerCapabilities <positive_integer>] [-appflowLog ( ENABLED | DISABLED )] [-netProfile <string>] [-
icmpVsrResponse ( PASSIVE | ACTIVE )] [-RHlstate ( PASSIVE | ACTIVE )] [-authnProfile <string>]
```

Arguments

name

Name for the content switching virtual server. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at sign (@), equal sign (=), and hyphen (-) characters.

Cannot be changed after the CS virtual server is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, \?my server\? or \?my server\?).

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

serviceType

Protocol used by the virtual server.

Possible values: HTTP, SSL, TCP, FTP, RTSP, SSL_TCP, UDP, DNS, SIP_UDP, ANY, RADIUS, RDP, MYSQL, MSSQL, DIAMETER, SSL_DIAMETER, DNS_TCP, ORACLE

IPAddress

IP address of the content switching virtual server.

IPPattern

IP address pattern, in dotted decimal notation, for identifying packets to be accepted by the virtual server. The IP Mask parameter specifies which part of the destination IP address is matched against the pattern. Mutually exclusive with the IP Address parameter.

For example, if the IP pattern assigned to the virtual server is 198.51.100.0 and the IP mask is 255.255.240.0 (a forward mask), the first 20 bits in the destination IP addresses are matched with the first 20 bits in the pattern. The virtual server accepts requests with IP addresses that range from 198.51.96.1 to 198.51.111.254. You can also use a pattern such as 0.0.2.2 and a mask such as 0.0.255.255 (a reverse mask).

If a destination IP address matches more than one IP pattern, the pattern with the longest match is selected, and the associated virtual server processes the request. For example, if the virtual servers, vs1 and vs2, have the same IP pattern, 0.0.100.128, but different IP masks of 0.0.255.255 and 0.0.224.255, a destination IP address of 198.51.100.128 has the longest match with the IP pattern of vs1. If a destination IP address matches two or more virtual servers to the same extent, the request is processed by the virtual server whose port number matches the port number in the request.

IPMask

IP mask, in dotted decimal notation, for the IP Pattern parameter. Can have leading or trailing non-zero octets (for example, 255.255.240.0 or 0.0.255.255). Accordingly, the mask specifies whether the first n bits or the last n bits of the destination IP address in a client request are to be matched with the corresponding bits in the IP pattern. The former is called a forward mask. The latter is called a reverse mask.

range

Number of consecutive IP addresses, starting with the address specified by the IP Address parameter, to include in a range of addresses assigned to this virtual server.

Default value: 1

Minimum value: 1

Maximum value: 254

port

Port number for content switching virtual server.

Minimum value: 1

state

Initial state of the load balancing virtual server.

Possible values: ENABLED, DISABLED

Default value: ENABLED

stateupdate

Enable state updates for a specific content switching virtual server. By default, the Content Switching virtual server is always UP, regardless of the state of the Load Balancing virtual servers bound to it. This parameter interacts with the global setting as follows:

Global Level | Vserver Level | Result

ENABLED	ENABLED	ENABLED
ENABLED	DISABLED	ENABLED
DISABLED	ENABLED	ENABLED
DISABLED	DISABLED	DISABLED

If you want to enable state updates for only some content switching virtual servers, be sure to disable the state update parameter.

Possible values: ENABLED, DISABLED

Default value: DISABLED

cacheable

Use this option to specify whether a virtual server, used for load balancing or content switching, routes requests to the cache redirection virtual server before sending it to the configured servers.

Possible values: YES, NO

Default value: NO

redirectURL

URL to which traffic is redirected if the virtual server becomes unavailable. The service type of the virtual server should be either HTTP or SSL.

Caution: Make sure that the domain in the URL does not match the domain specified for a content switching policy. If it does, requests are continuously redirected to the unavailable virtual server.

cltTimeout

Idle time, in seconds, after which the client connection is terminated. The default values are:

180 seconds for HTTP/SSL-based services.

9000 seconds for other TCP-based services.

120 seconds for DNS-based services.

120 seconds for other UDP-based services.

Default value: -1

Maximum value: 31536000

precedence

Type of precedence to use for both RULE-based and URL-based policies on the content switching virtual server. With the default (RULE) setting, incoming requests are evaluated against the rule-based content switching policies. If none of the rules match, the URL in the request is evaluated against the URL-based content switching policies.

Possible values: RULE, URL

Default value: RULE

caseSensitive

Consider case in URLs (for policies that use URLs instead of RULES). For example, with the ON setting, the URLs /a/1.html and /A/1.HTML are treated differently and can have different targets (set by content switching policies). With the OFF setting, /a/1.html and /A/1.HTML are switched to the same target.

Possible values: ON, OFF

Default value: ON

soMethod

Type of spillover used to divert traffic to the backup virtual server when the primary virtual server reaches the spillover threshold. Connection spillover is based on the number of connections. Bandwidth spillover is based on the total Kbps of incoming and outgoing traffic.

Possible values: CONNECTION, DYNAMICCONNECTION, BANDWIDTH, HEALTH, NONE

soPersistence

Maintain source-IP based persistence on primary and backup virtual servers.

Possible values: ENABLED, DISABLED

Default value: DISABLED

soPersistenceTimeOut

Time-out value, in minutes, for spillover persistence.

Default value: 2

Minimum value: 2

Maximum value: 1440

soThreshold

Depending on the spillover method, the maximum number of connections or the maximum total bandwidth (Kbps) that a virtual server can handle before spillover occurs.

Minimum value: 1

Maximum value: 4294967287

soBackupAction

Action to be performed if spillover is to take effect, but no backup chain to spillover is usable or exists

Possible values: DROP, ACCEPT, REDIRECT

redirectPortRewrite

State of port rewrite while performing HTTP redirect.

Possible values: ENABLED, DISABLED

Default value: DISABLED

downStateFlush

Flush all active transactions associated with a virtual server whose state transitions from UP to DOWN. Do not enable this option for applications that must complete their transactions.

Possible values: ENABLED, DISABLED

Default value: ENABLED

backupVServer

Name of the backup virtual server that you are configuring. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at sign (@), equal sign (=), and hyphen (-) characters. Can be changed after the backup virtual server is created. You can assign a different backup virtual server or rename the existing virtual server.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks.

disablePrimaryOnDown

Continue forwarding the traffic to backup virtual server even after the primary server comes UP from the DOWN state.

Possible values: ENABLED, DISABLED

Default value: DISABLED

insertVserverIPPort

Insert the virtual server's VIP address and port number in the request header. Available values function as follows:

VIPADDR - Header contains the vserver's IP address and port number without any translation.

OFF - The virtual IP and port header insertion option is disabled.

V6TOV4MAPPING - Header contains the mapped IPv4 address corresponding to the IPv6 address of the vserver and the port number. An IPv6 address can be mapped to a user-specified IPv4 address using the set ns ip6 command.

Possible values: OFF, VIPADDR, V6TOV4MAPPING

vipHeader

Name of virtual server IP and port header, for use with the VServer IP Port Insertion parameter.

rtspNat

Enable network address translation (NAT) for real-time streaming protocol (RTSP) connections.

Possible values: ON, OFF

Default value: OFF

AuthenticationHost

FQDN of the authentication virtual server. The service type of the virtual server should be either HTTP or SSL.

Authentication

Authenticate users who request a connection to the content switching virtual server.

Possible values: ON, OFF

Default value: OFF

Listenpolicy

String specifying the listen policy for the content switching virtual server. Can be either the name of an existing expression or an in-line expression.

Default value: "none"

Listenpriority

Integer specifying the priority of the listen policy. A higher number specifies a lower priority. If a request matches the listen policies of more than one virtual server the virtual server whose listen policy has the highest priority (the lowest priority number) accepts the request.

Default value: 101

Minimum value: 0

Maximum value: 100

authn401

Enable HTTP 401-response based authentication.

Possible values: ON, OFF

Default value: OFF

authnVsName

Name of authentication virtual server that authenticates the incoming user requests to this content switching virtual server.

push

Process traffic with the push virtual server that is bound to this content switching virtual server (specified by the Push VServer parameter). The service type of the push virtual server should be either HTTP or SSL.

Possible values: ENABLED, DISABLED

Default value: DISABLED

pushVserver

Name of the load balancing virtual server, of type PUSH or SSL_PUSH, to which the server pushes updates received on the client-facing load balancing virtual server.

pushLabel

Expression for extracting the label from the response received from server. This string can be either an existing rule name or an inline expression. The service type of the virtual server should be either HTTP or SSL.

Default value: "none"

pushMultiClients

Allow multiple Web 2.0 connections from the same client to connect to the virtual server and expect updates.

Possible values: YES, NO

Default value: NO

tcpProfileName

Name of the TCP profile containing TCP configuration settings for the virtual server.

httpProfileName

Name of the HTTP profile containing HTTP configuration settings for the virtual server. The service type of the virtual server should be either HTTP or SSL.

dbProfileName

Name of the DB profile.

oracleServerVersion

Oracle server version

Possible values: 10G, 11G

Default value: 10G

comment

Information about this virtual server.

mssqlServerVersion

The version of the MSSQL server

Possible values: 70, 2000, 2000SP1, 2005, 2008, 2008R2, 2012

Default value: 2008R2

l2Conn

Use L2 Parameters to identify a connection

Possible values: ON, OFF

mysqlProtocolVersion

The protocol version returned by the mysql vserver.

Default value: 10

Minimum value: 0

mysqlServerVersion

The server version string returned by the mysql vserver.

Default value: NSA_MYSQL_SERVER_VER_DEFAULT

mysqlCharacterSet

The character set returned by the mysql vserver.

Default value: 8

Minimum value: 0

mysqlServerCapabilities

The server capabilities returned by the mysql vserver.

Default value: 41613

Minimum value: 0

appflowLog

Enable logging appflow flow information

Possible values: ENABLED, DISABLED

Default value: ENABLED

netProfile

The name of the network profile.

icmpVsrResponse

Can be active or passive

Possible values: PASSIVE, ACTIVE

Default value: PASSIVE

RHlstate

A host route is injected according to the setting on the virtual servers

* If set to PASSIVE on all the virtual servers that share the IP address, the appliance always injects the hostroute.

* If set to ACTIVE on all the virtual servers that share the IP address, the appliance injects even if one virtual server is UP.

* If set to ACTIVE on some virtual servers and PASSIVE on the others, the appliance, injects even if one virtual server set to ACTIVE is UP.

Possible values: PASSIVE, ACTIVE

Default value: PASSIVE

authnProfile

Name of the authentication profile to be used when authentication is turned on.

Example

1. You can use precedence when certain client attributes (e.g., browser type) require to l

rm cs vserver

Removes a content switching virtual server.

Synopsys

```
rm cs vserver <name>@ ...
```

Arguments

name

Name of the virtual server to be removed.

Example

```
rm vserver cs_vip
```

set cs vserver

Modifies the configuration of a content switching virtual server.

Synopsys

```
set cs vserver <name> [-IPAddress <ip_addr|ipv6_addr|*>] [-IPPattern <ippat>] [-IPMask <ipmask>] [-stateupdate (
ENABLED | DISABLED )] [-precedence ( RULE | URL )] [-caseSensitive ( ON | OFF )] [-backupVServer <string>] [-
redirectURL <URL>] [-cacheable ( YES | NO )] [-cltTimeout <secs>] [-soMethod <soMethod>] [-soPersistence (
ENABLED | DISABLED )] [-soPersistenceTimeout <positive_integer>] [-soThreshold <positive_integer>] [-
soBackupAction <soBackupAction>] [-redirectPortRewrite ( ENABLED | DISABLED )] [-downStateFlush ( ENABLED
| DISABLED )] [-disablePrimaryOnDown ( ENABLED | DISABLED )] [-insertVserverIPPort <insertVserverIPPort>
<vipHeader>] [-rtspNat ( ON | OFF )] [-AuthenticationHost <string>] [-Authentication ( ON | OFF )] [-Listenpolicy
<expression>] [-Listenpriority <positive_integer>] [-authn401 ( ON | OFF )] [-authnVsName <string>] [-push (
ENABLED | DISABLED )] [-pushVserver <string>] [-pushLabel <expression>] [-pushMultiClients ( YES | NO )] [-
tcpProfileName <string>] [-httpProfileName <string>] [-dbProfileName <string>] [-comment <string>] [-l2Conn ( ON |
OFF )] [-mssqlServerVersion <mssqlServerVersion>] [-mysqlProtocolVersion <positive_integer>] [-
oracleServerVersion ( 10G | 11G )] [-mysqlServerVersion <string>] [-mysqlCharacterSet <positive_integer>] [-
mysqlServerCapabilities <positive_integer>] [-appflowLog ( ENABLED | DISABLED )] [-netProfile <string>] [-
authnProfile <string>] [-icmpVsrResponse ( PASSIVE | ACTIVE )] [-RHlstate ( PASSIVE | ACTIVE )]
```

Arguments

name

Identifies the virtual server name (created with the add cs vserver command).

IPAddress

The new IP address of the virtual server.

IPPattern

IP address pattern, in dotted decimal notation, for identifying packets to be accepted by the virtual server. The IP Mask parameter specifies which part of the destination IP address is matched against the pattern. Mutually exclusive with the IP Address parameter.

For example, if the IP pattern assigned to the virtual server is 198.51.100.0 and the IP mask is 255.255.240.0 (a forward mask), the first 20 bits in the destination IP addresses are matched with the first 20 bits in the pattern. The virtual server accepts requests with IP addresses that range from 198.51.96.1 to 198.51.111.254. You can also use a pattern such as 0.0.2.2 and a mask such as 0.0.255.255 (a reverse mask).

If a destination IP address matches more than one IP pattern, the pattern with the longest match is selected, and the associated virtual server processes the request. For example, if the virtual servers, vs1 and vs2, have the same IP pattern, 0.0.100.128, but different IP masks of 0.0.255.255 and 0.0.224.255, a destination IP address of 198.51.100.128 has the longest match with the IP pattern of vs1. If a destination IP address matches two or more virtual servers to the same extent, the request is processed by the virtual server whose port number matches the port number in the request.

IPMask

IP mask, in dotted decimal notation, for the IP Pattern parameter. Can have leading or trailing non-zero octets (for example, 255.255.240.0 or 0.0.255.255). Accordingly, the mask specifies whether the first n bits or the last n bits of the destination IP address in a client request are to be matched with the corresponding bits in the IP pattern. The former is called a forward mask. The latter is called a reverse mask.

stateupdate

Enable state updates for a specific content switching virtual server. By default, the Content Switching virtual server is always UP, regardless of the state of the Load Balancing virtual servers bound to it. This parameter interacts with the global setting as follows:

Global Level | Vserver Level | Result

ENABLED	ENABLED	ENABLED
ENABLED	DISABLED	ENABLED
DISABLED	ENABLED	ENABLED
DISABLED	DISABLED	DISABLED

If you want to enable state updates for only some content switching virtual servers, be sure to disable the state update parameter.

Possible values: ENABLED, DISABLED

Default value: DISABLED

precedence

The precedence on the content switching virtual server between rule-based and URL-based policies. The default precedence is set to RULE.

If the precedence is configured as RULE, the incoming request is applied against the content switching policies created with the -rule argument. If none of the rules match, then the URL in the request is applied against the content switching policies created with the -url option.

For example, this precedence can be used if certain client attributes (such as a specific type of browser) need to be served different content and all other clients can be served from the content distributed among the servers.

If the precedence is configured as URL, the incoming request URL is applied against the content switching policies created with the -url option. If none of the policies match, then the request is applied against the content switching policies created with the -rule option.

Also, this precedence can be used if some content (such as images) is the same for all clients, but other content (such as text) is different for different clients. In this case, the images will be served to all clients, but the text will be served to specific clients based on specific attributes, such as Accept-Language.

Possible values: RULE, URL

Default value: RULE

caseSensitive

The URL lookup case option on the content switching vserver.

If case sensitivity of a content switching virtual server is set to 'ON', the URLs /a/1.html and /A/1.HTML are treated differently and may have different targets (set by content switching policies).

If case sensitivity is set to 'OFF', the URLs /a/1.html and /A/1.HTML are treated the same, and will be switched to the same target.

Possible values: ON, OFF

Default value: ON

backupVServer

Name of the backup virtual server that you are configuring. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at sign (@), equal sign (=), and hyphen (-) characters. Can be changed after the backup virtual server is created. You can assign a different backup virtual server or rename the existing virtual server.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks.

redirectURL

The redirect URL for content switching.

cacheable

The option to specify whether a virtual server used for content switching will route requests to the cache redirection virtual server before sending it to the configured servers.

Possible values: YES, NO

Default value: NO

cltTimeout

Client timeout in seconds.

Default value: -1

Maximum value: 31536000

soMethod

The spillover factor. When traffic on the main virtual server reaches this threshold, additional traffic is sent to the backupvserver.

Possible values: CONNECTION, DYNAMICCONNECTION, BANDWIDTH, HEALTH, NONE

soPersistence

Maintain source-IP based persistence on primary and backup virtual servers.

Possible values: ENABLED, DISABLED

Default value: DISABLED

soPersistenceTimeOut

The spillover persistency entry timeout.

Default value: 2

Minimum value: 2

Maximum value: 1440

soThreshold

Depending on the spillover method, the maximum number of connections or the maximum total bandwidth (Kbps) that a virtual server can handle before spillover occurs.

Minimum value: 1

Maximum value: 4294967287

soBackupAction

Action to be performed if spillover is to take effect, but no backup chain to spillover is usable or exists

Possible values: DROP, ACCEPT, REDIRECT

redirectPortRewrite

SSL redirect port rewrite.

Possible values: ENABLED, DISABLED

Default value: DISABLED

downStateFlush

Flush all active transactions associated with a virtual server whose state transitions from UP to DOWN. Do not enable this option for applications that must complete their transactions.

Possible values: ENABLED, DISABLED

Default value: ENABLED

disablePrimaryOnDown

Continue forwarding the traffic to backup virtual server even after the primary server comes UP from the DOWN state.

Possible values: ENABLED, DISABLED

Default value: DISABLED

insertVserverIPPort

The virtual IP and port header insertion option for the vserver.

* VIPADDR - Header contains the vserver's IP address and port number without any translation.

* OFF - The virtual IP and port header insertion option is disabled.

* V6TOV4MAPPING - Header contains the mapped IPv4 address that corresponds to the IPv6 address of the vserver and the port number. An IPv6 address can be mapped to a user-specified IPv4 address using the set ns ip6 command.

Possible values: OFF, VIPADDR, V6TOV4MAPPING

vipHeader

The name of virtual IP and port header.

rtspNat

Enable network address translation (NAT) for real-time streaming protocol (RTSP) connections.

Possible values: ON, OFF

Default value: OFF

AuthenticationHost

FQDN of the authentication virtual server. The service type of the virtual server should be either HTTP or SSL.

Authentication

Authenticate users who request a connection to the content switching virtual server.

Possible values: ON, OFF

Default value: OFF

Listenpolicy

String specifying the listen policy for the content switching virtual server. Can be either the name of an existing expression or an in-line expression.

Default value: "none"

Listenpriority

Integer specifying the priority of the listen policy. A higher number specifies a lower priority. If a request matches the listen policies of more than one virtual server the virtual server whose listen policy has the highest priority (the lowest priority number) accepts the request.

Default value: 101

Minimum value: 0

Maximum value: 100

authn401

Enable HTTP 401-response based authentication.

Possible values: ON, OFF

Default value: OFF

authnVsName

Name of authentication virtual server that authenticates the incoming user requests to this content switching virtual server.

push

Process traffic with the push virtual server that is bound to this content switching virtual server (specified by the Push VServer parameter). The service type of the push virtual server should be either HTTP or SSL.

Possible values: ENABLED, DISABLED

Default value: DISABLED

pushVserver

Name of the load balancing virtual server, of type PUSH or SSL_PUSH, to which the server pushes updates received on the client-facing load balancing virtual server.

pushLabel

Expression for extracting the label from the response received from server. This string can be either an existing rule name or an inline expression. The service type of the virtual server should be either HTTP or SSL.

Default value: "none"

pushMultiClients

Allow multiple Web 2.0 connections from the same client to connect to the virtual server and expect updates.

Possible values: YES, NO

Default value: NO

tcpProfileName

Name of the TCP profile containing TCP configuration settings for the virtual server.

httpProfileName

Name of the HTTP profile containing HTTP configuration settings for the virtual server. The service type of the virtual server should be either HTTP or SSL.

dbProfileName

Name of the DB profile.

comment

Information about this virtual server.

l2Conn

Use L2 Parameters to identify a connection

Possible values: ON, OFF

mssqlServerVersion

The version of the MSSQL server

Possible values: 70, 2000, 2000SP1, 2005, 2008, 2008R2, 2012

Default value: 2008R2

mysqlProtocolVersion

The protocol version returned by the mysql vserver.

Default value: 10

Minimum value: 0

oracleServerVersion

Oracle server version

Possible values: 10G, 11G

Default value: 10G

mysqlServerVersion

The server version string returned by the mysql vserver.

Default value: NSA_MYSQL_SERVER_VER_DEFAULT

mysqlCharacterSet

The character set returned by the mysql vserver.

Default value: 8

Minimum value: 0

mysqlServerCapabilities

The server capabilities returned by the mysql vserver.

Default value: 41613

Minimum value: 0

appflowLog

Enable logging appflow flow information

Possible values: ENABLED, DISABLED

Default value: ENABLED

netProfile

The name of the network profile.

authnProfile

Name of the authentication profile to be used when authentication is turned on.

icmpVsrResponse

Can be active or passive

Possible values: PASSIVE, ACTIVE

Default value: PASSIVE

RHlstate

A host route is injected according to the setting on the virtual servers

* If set to PASSIVE on all the virtual servers that share the IP address, the appliance always injects the hostroute.

* If set to ACTIVE on all the virtual servers that share the IP address, the appliance injects even if one virtual server is UP.

* If set to ACTIVE on some virtual servers and PASSIVE on the others, the appliance, injects even if one virtual server set to ACTIVE is UP.

Possible values: PASSIVE, ACTIVE

Default value: PASSIVE

unset cs vserver

Unset the parameters of a content switching virtual server..Refer to the set cs vserver command for meanings of the arguments.

Synopsis

```
unset cs vserver <name> [-caseSensitive] [-backupVServer] [-cltTimeout] [-redirectURL] [-authn401] [-Authentication] [-AuthenticationHost] [-authnVsName] [-pushVserver] [-pushLabel] [-tcpProfileName] [-httpProfileName] [-dbProfileName] [-l2Conn] [-mysqlProtocolVersion] [-mysqlServerVersion] [-mysqlCharacterSet] [-mysqlServerCapabilities] [-appflowLog] [-netProfile] [-icmpVsrResponse] [-authnProfile] [-stateupdate] [-precedence] [-cacheable] [-soMethod] [-soPersistence] [-soPersistenceTimeOut] [-soThreshold] [-soBackupAction] [-redirectPortRewrite] [-downStateFlush] [-disablePrimaryOnDown] [-insertVserverIPPort] [-vipHeader] [-rtspNat] [-Listenpolicy] [-Listenpriority] [-push] [-pushMultiClients] [-comment] [-mssqlServerVersion] [-oracleServerVersion] [-RHlstate]
```

bind cs vserver

Binds a content switching virtual server to a content switching policy.

Synopsis

```
bind cs vserver <name> [-lbvserver <string> | (-policyName <string> [-targetLBVserver <string>] [-priority <positive_integer>] [-gotoPriorityExpression <expression>] [-type ( REQUEST | RESPONSE )] [-invoke (<labelType> <labelName> ) ] ]]
```

Arguments

name

Name of the content switching virtual server to which the content switching policy applies.

lbvserver

Name of the default Load Balancing vserver bound. If for a particular content none of the Content Switching policies is evaluated to TRUE, that traffic is switched to default Load Balancing vserver. .

Example: bind cs vserver cs1 -lbvserver lb1

Note: Use this parameter for default binding only.

policyName

Name of the content switching policy to bind to the content switching virtual server. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at sign (@), equal sign (=), and hyphen (-) characters. Cannot be changed after a policy is created.

To bind a content switching policy, you need a content-based virtual server (content switching virtual server) and an address-based virtual server (load balancing virtual server). You can assign multiple policies to the virtual server pair.

Note: When binding a CS virtual server to a default LB virtual server, the Policy Name parameter is optional.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my policy" or 'my policy').

targetLBVserver

Name of the Load Balancing virtual server to which the content is switched, if policy rule is evaluated to be TRUE.

Example: bind cs vs cs1 -policyname pol1 -priority 101 -targetLBVserver lb1

Note: Use this parameter only in case of Content Switching policy bind operations to a CS vserver

priority

Unsigned integer that determines the priority of the policy relative to other policies in this policy label (bound to the same bind point). A lower number specifies a higher priority. Priority cannot be specified if the Content Switching policy is URL based. The maximum value of priority for a default-syntax content switching policy is 2147483647. The maximum value for a classic content switching policy is 4294967295.

Minimum value: 0

gotoPriorityExpression

Expression or other value specifying the next policy to be evaluated if the current policy evaluates to TRUE. Specify one of the following values:

- * NEXT ? Evaluate the policy with the next higher priority number.
- * END ? End policy evaluation.
- * USE_INVOCATION_RESULT ? Applicable if this policy invokes another policy label. If the final goto in the invoked policy label has a value of END, the evaluation stops. If the final goto is anything other than END, the current policy label performs a NEXT.
- * A default syntax expression that evaluates to a number.

If you specify an expression, the number to which it evaluates determines the next policy to evaluate, as follows:

- * If the expression evaluates to a higher numbered priority, the policy with that priority is evaluated next.
- * If the expression evaluates to the priority of the current policy, the policy with the next higher numbered priority is evaluated next.
- * If the expression evaluates to a priority number that is numerically higher than the highest numbered priority, policy evaluation ends.

An UNDEF event is triggered if:

- * The expression is invalid.
- * The expression evaluates to a priority number that is numerically lower than the current policy's priority.
- * The expression evaluates to a priority number that is between the current policy's priority number (say, 30) and the highest priority number (say, 100), but does not match any configured priority number (for example, the expression evaluates to the number 85). This example assumes that the priority number increments by 10 for every successive policy, and therefore a priority number of 85 does not exist in the policy label.

type

For a rewrite policy, the bind point to which to bind the policy. Note: This parameter applies only to rewrite policies, because content switching policies are evaluated only at request time.

Possible values: REQUEST, RESPONSE

invoke

Invoke a policy label if this policy's rule evaluates to TRUE (valid only for default-syntax policies such as application firewall, transform, integrated cache, rewrite, responder, and content switching).

labelType

Type of label to be invoked.

Possible values: reqvserver, resvserver, policylabel

labelName

Name of the label to be invoked.

Example

```
i) bind cs vserver csw-vip1 -policyname csw-policy1 -priority 13 ii) bind cs vserver
```

unbind cs vserver

Unbinds the virtual server from the content switching policy.

Synopsys

```
unbind cs vserver <name> [(-policyName <string> [-type ( REQUEST | RESPONSE )]) | -lbvserver <string>] [-priority <positive_integer>]
```

Arguments

name

Name of the virtual server to unbind from the policy.

policyName

Name of the policy from which to unbind the content switching virtual server. Note: To unbind the content switching virtual server from the default policy, do not specify a value for this parameter.

type

For rewrite policies, the traffic flow to which the policy applies. Note: This parameter applies only to rewrite policies, because content switching policies are evaluated only at request time.

Possible values: REQUEST, RESPONSE

priority

Priority number of the policy from which to unbind the content switching virtual server.

Minimum value: 1

lbvserver

The virtual server name (created with the add lb vserver command) to which content will be switched.

Default value: "default_lb"

enable cs vserver

Enables a content switching virtual server.

Synopsys

```
enable cs vserver <name>@
```

Arguments

name

Name of the content switching virtual server to enable.

Note: Virtual servers, when added, are enabled by default.

Example

```
enable vserver cs_vip
```

disable cs vserver

Disables a content switching virtual server.

Synopsys

```
disable cs vserver <name>@
```

Arguments

name

Name of the virtual server to be disabled.

Example

```
disable vserver cs_vip
```

show cs vserver

Displays all existing content switching virtual servers, or just the specified virtual server.

Synopsys

```
show cs vserver [<name>] show cs vserver stats - alias for 'stat cs vserver'
```

Arguments

name

Name of a content switching virtual server for which to display information, including the policies bound to the virtual server. To display a list of all configured Content Switching virtual servers, do not specify a value for this parameter.

Outputs

insertVserverIPPort

The virtual IP and port header insertion option for the vserver.

vipHeader

The name of virtual IP and port header.

IPAddress

IP address of the content switching virtual server.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

IPPattern

The IP address of the virtual server.

IPMask

The IP address mask of the virtual server.

stateflag

value

The ssl card status for the transparent ssl cs vserver.

port

Port number for content switching virtual server.

range

Number of consecutive IP addresses, starting with the address specified by the IP Address parameter, to include in a range of addresses assigned to this virtual server.

serviceType

Protocol used by the virtual server.

ngname

Nodegroup devno to which this csvserver belongs to

type

The bindpoint to which the policy is bound

vsvrcfgflags

Contains the config info of vserver to be used at validation

state

Initial state of the load balancing virtual server.

sc

The state of SureConnect the specified virtual server.

stateupdate

Enable state updates for a specific content switching virtual server. By default, the Content Switching virtual server is always UP, regardless of the state of the Load Balancing virtual servers bound to it. This parameter interacts with the global setting as follows:

Global Level | Vserver Level | Result

ENABLED	ENABLED	ENABLED
ENABLED	DISABLED	ENABLED
DISABLED	ENABLED	ENABLED
DISABLED	DISABLED	DISABLED

If you want to enable state updates for only some content switching virtual servers, be sure to disable the state update parameter.

status

Status.

cacheType

Cache type.

redirect

Redirect URL string.

precedence

Type of precedence to use for both RULE-based and URL-based policies on the content switching virtual server. With the default (RULE) setting, incoming requests are evaluated against the rule-based content switching policies. If none of the rules match, the URL in the request is evaluated against the URL-based content switching policies.

redirectURL

The redirect URL for content switching.

Authentication

Authentication.

authn401

HTTP 401 response based authentication.

authnVsName

Name of authentication virtual server that authenticates the incoming user requests to this content switching virtual server.

caseSensitive

Consider case in URLs (for policies that use URLs instead of RULES). For example, with the ON setting, the URLs /a/1.html and /A/1.HTML are treated differently and can have different targets (set by content switching policies). With the OFF setting, /a/1.html and /A/1.HTML are switched to the same target.

homePage

Home page.

dnsVserverName

DNS vserver name.

domain

Domain.

rule

Rule.

policyName

Policies bound to this vserver.

hits

Number of hits.

piPolicyhits

Number of hits.

serviceName

Service name.

weight

Weight for this service.

cacheVserver

Cache vserver name.

targetVserver

target vserver name.

backupVServer

Name of the backup virtual server that you are configuring. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at sign (@), equal sign (=), and hyphen (-) characters. Can be changed after the backup virtual server is created. You can assign a different backup virtual server or rename the existing virtual server.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks.

priority

Priority for the policy.

cltTimeout

Idle time, in seconds, after which the client connection is terminated. The default values are:

180 seconds for HTTP/SSL-based services.

9000 seconds for other TCP-based services.

120 seconds for DNS-based services.

120 seconds for other UDP-based services.

Listenpolicy

The string is listenpolicy configured for lb vserver

Listenpriority

This parameter is the priority for listen policy of LB Vserver.

soMethod

Type of spillover used to divert traffic to the backup virtual server when the primary virtual server reaches the spillover threshold. Connection spillover is based on the number of connections. Bandwidth spillover is based on the total Kbps of incoming and outgoing traffic.

soPersistence

Maintain source-IP based persistence on primary and backup virtual servers.

soPersistenceTimeOut

Time-out value, in minutes, for spillover persistence.

soThreshold

Depending on the spillover method, the maximum number of connections or the maximum total bandwidth (Kbps) that a virtual server can handle before spillover occurs.

soBackupAction

Action to be performed if spillover is to take effect, but no backup chain to spillover is usable or exists

cacheable

The state of caching.

url

URL string.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

redirectPortRewrite

Redirect port rewrite.

downStateFlush

Flush all active transactions associated with a virtual server whose state transitions from UP to DOWN. Do not enable this option for applications that must complete their transactions.

disablePrimaryOnDown

Tells whether traffic will continue reaching backup vservers even after primary comes UP from DOWN state.

invoke

Invoke flag.

labelType

The invocation type.

labelName

Name of the label invoked.

gt2GB

This argument has no effect.

stateChangeTimeSec

Time when last state change happened. Seconds part.

stateChangeTimeSec

Time at which last state change happened. Milliseconds part.

ticksSinceLastStateChange

Time in 10 millisecond ticks since the last state change.

rtspNat

Enable network address translation (NAT) for real-time streaming protocol (RTSP) connections.

AuthenticationHost

FQDN of the authentication virtual server. The service type of the virtual server should be either HTTP or SSL.

push

Process traffic with the push virtual server that is bound to this content switching virtual server (specified by the Push VServer parameter). The service type of the push virtual server should be either HTTP or SSL.

pushVserver

Name of the load balancing virtual server, of type PUSH or SSL_PUSH, to which the server pushes updates received on the client-facing load balancing virtual server.

pushLabel

Expression for extracting the label from the response received from server. This string can be either an existing rule name or an inline expression. The service type of the virtual server should be either HTTP or SSL.

pushMultiClients

Allow multiple Web 2.0 connections from the same client to connect to the virtual server and expect updates.

tcpProfileName

Name of the TCP profile containing TCP configuration settings for the virtual server.

httpProfileName

Name of the HTTP profile containing HTTP configuration settings for the virtual server. The service type of the virtual server should be either HTTP or SSL.

dbProfileName

Name of the DB profile.

comment

Information about this virtual server.

appfwPolicyFlag**flags****policySubType****oracleServerVersion**

Oracle server version

mssqlServerVersion

The version of the MSSQL server

l2Conn

Use L2 Parameters to identify a connection

mysqlProtocolVersion

The protocol version returned by the mysql vserver.

mysqlServerVersion

The server version string returned by the mysql vserver.

mysqlCharacterSet

The character set returned by the mysql vserver.

mysqlServerCapabilities

The server capabilities returned by the mysql vserver.

appflowLog

Enable logging appflow flow information

netProfile

The name of the network profile.

icmpVsrResponse

Can be active or passive

RHIstate

A host route is injected according to the setting on the virtual servers

* If set to PASSIVE on all the virtual servers that share the IP address, the appliance always injects the hostroute.

* If set to ACTIVE on all the virtual servers that share the IP address, the appliance injects even if one virtual server is UP.

* If set to ACTIVE on some virtual servers and PASSIVE on the others, the appliance, injects even if one virtual server set to ACTIVE is UP.

lbvserver

Name of the default lb vserver bound. Use this param for Default binding only. For Example: bind cs vserver cs1 -lbvserver lb1

targetLBVserver

target vserver name.

contentVsvrFlag**authnProfile**

Name of the authentication profile to be used when authentication is turned on.

devno**count**

stat cs vserver

Displays statistics of all content switching virtual servers, or statistics for just the specified content switching virtual server.

Synopsys

```
stat cs vserver [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full )]
```

Arguments

name

Name of the content switching virtual server for which to display statistics. To display statistics for all configured Content Switching virtual servers, do not specify a value for this parameter.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Current Client Est connections (CIntEstConn)

Number of client connections in ESTABLISHED state.

IP address (IP)

The IP address on which the service is running.

Port (port)

The port on which the service is running.

Vserver protocol (Protocol)

Protocol associated with the vservers

State

Current state of the server. Possible values are UP, DOWN, UNKNOWN, OFS(Out of Service), TROFS (Transition Out of Service), TROFS_DOWN(Down When going Out of Service)

Vserver hits (Hits)

Total vservers hits

Requests (Req)

Total number of requests received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

Responses (Rsp)

Number of responses received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

Request bytes (Reqb)

Total number of request bytes received on this service or virtual server.

Response bytes (Rspb)

Number of response bytes received by this service or virtual server.

Total Packets rcvd (PktRx)

Total number of packets received by this service or virtual server.

Total Packets sent (PktTx)

Total number of packets sent.

Current client connections (CIntConn)

Number of current client connections.

Current server connections (SvrConn)

Number of current connections to the actual servers behind the virtual server.

Spill Over Threshold (SOTresh)

Spill Over Threshold set on the VServer.

Spill Over Hits (NumSo)

Number of times vserver experienced spill over.

Labeled Connection (LblConn)

Number of Labeled connection on this vserver

Push Labeled Connection (PushLbl)

Number of labels for this push vserver.

Deferred Request (DefReq)

Number of deferred request on this vserver

Invalid Request/Response (IvldReqRsp)

Number invalid requests/responses on this vserver

Invalid Request/Response Dropped (IvldReqRspDrp)

Number invalid requests/responses dropped on this vserver

Current Server Est connections (SvrEstConn)

Number of server connections in ESTABLISHED state.

rename cs vserver

Renames a content switching virtual server.

Synopsis

rename cs vserver <name>@ <newName>@

Arguments

name

Existing name of the content switching virtual server.

newName

New name for the virtual server. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at sign (@), equal sign (=), and hyphen (-) characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my name" or 'my name').

Example

```
rename cs vserver cs1 cs2
```


DB Commands

The entities on which you can perform NetScaler CLI operations:

- db dbProfile
- db user

db dbProfile

The following operations can be performed on "db dbProfile":

add | **rm** | **set** | **unset** | **show**

add db dbProfile

Add a new DB profile on the Netscaler

Synopsys

```
add db dbProfile <name> [-interpretQuery ( YES | NO )] [-stickiness ( YES | NO )] [-kcdAccount <string>] [-conMultiplex ( ENABLED | DISABLED )] [-enableCachingConMuxOFF ( ENABLED | DISABLED )]
```

Arguments

name

Name for the database profile. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at sign (@), equal sign (=), and hyphen (-) characters. Cannot be changed after the profile is created.

CLI Users: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my profile" or 'my profile').

interpretQuery

If ENABLED, inspect the query and update the connection information, if required. If DISABLED, forward the query to the server.

Possible values: YES, NO

Default value: YES

stickiness

If the queries are related to each other, forward to the same backend server.

Possible values: YES, NO

Default value: NO

kcdAccount

Name of the KCD account that is used for Windows authentication.

conMultiplex

Use the same server-side connection for multiple client-side requests. Default is enabled.

Possible values: ENABLED, DISABLED

Default value: ENABLED

enableCachingConMuxOFF

Enable caching when connection multiplexing is OFF.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
add dbprofile <profile name> -interpretQuery YES -stickiness YES -kcdaccount account
```

rm db dbProfile

Remove a DB profile on the Netscaler

Synopsys

```
rm db dbProfile <name>
```

Arguments

name

Name of the DB profile

Example

```
rm dbprofile <profile name>
```

set db dbProfile

Set/modify DB profile values

Synopsys

```
set db dbProfile <name> [-interpretQuery ( YES | NO )] [-stickiness ( YES | NO )] [-kcdAccount <string>] [-conMultiplex ( ENABLED | DISABLED )] [-enableCachingConMuxOFF ( ENABLED | DISABLED )]
```

Arguments

name

Name for the database profile. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at sign (@), equal sign (=), and hyphen (-) characters. Cannot be changed after the profile is created.

CLI Users: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my profile" or 'my profile').

interpretQuery

If ENABLED, inspect the query and update the connection information, if required. If DISABLED, forward the query to the server.

Possible values: YES, NO

Default value: YES

stickiness

If the queries are related to each other, forward to the same backend server.

Possible values: YES, NO

Default value: NO

kcdAccount

Name of the KCD account that is used for Windows authentication.

conMultiplex

Use the same server-side connection for multiple client-side requests. Default is enabled.

Possible values: ENABLED, DISABLED

Default value: ENABLED

enableCachingConMuxOFF

Enable caching when connection multiplexing is OFF.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
set dbprofile <profile name> -interpretQuery YES -stickyness YES
```

unset db dbProfile

Unset DB profile values. Refer to the set db dbProfile command for meanings of the arguments.

Synopsis

```
unset db dbProfile <name> [-interpretQuery] [-stickiness] [-kcdAccount] [-conMultiplex] [-enableCachingConMuxOFF]
```

show db dbProfile

Display all the configured DB profiles in the system. If a name is specified, then only that profile is shown.

Synopsis

```
show db dbProfile [<name>]
```

Arguments

name

Name of the DB profile.

Outputs

interpretQuery

Interpret Queries on NS

stickiness

Stickyness for Queries

kcdAccount

KCD account for windows authentication

conMultiplex

Enable/Disable Connection Multiplexing

refCnt

Profile Reference Count

enableCachingConMuxOFF

Enable Caching When Connection Multiplexing is OFF

stateflag

State flag

devno

count

Example

```
show dbprofile [profile name]
```

db user

The following operations can be performed on "db user":

[add](#) | [rm](#) | [set](#) | [show](#)

add db user

Adds a database user. The user name and password that you specify in this command are added to the nsconfig file and used to authenticate the user.

Synopsys

```
add db user <userName> {-password }
```

Arguments

userName

Name of the database user. Must be the same as the user name specified in the database.

password

Password for logging on to the database. Must be the same as the password specified in the database.

Example

```
add db user johndoe -password secret
```

rm db user

Removes a database user from the NetScaler appliance. Requests from the user are no longer authenticated or routed to the database server.

Synopsys

```
rm db user <userName>
```

Arguments

userName

Name of the database user to remove.

set db user

Modifies the password of an existing database user.

Synopsys

```
set db user <userName>
```

Arguments

userName

Name of the database user.

password

The database users password. If you use the CLI, you are prompted for this password after specifying the user name.

Example

```
set db user johndoe
```

 The above command sets the password for johndoe to abcd (Password to)

show db user

Displays the specified database user or, if no user is specified, all the database users configured on the appliance.

Synopsys

```
show db user [<userName>] [-loggedIn]
```

Arguments

userName

Name of the database user.

loggedIn

Display the names of all database users currently logged on to the NetScaler appliance.

Outputs

password

Password for logging on to the database. Must be the same as the password specified in the database.

devno

count

stateflag

DNS Commands

The entities on which you can perform NetScaler CLI operations:

- o dns
- o dns aaaaRec
- o dns action
- o dns action64
- o dns addRec
- o dns cnameRec
- o dns global
- o dns key
- o dns mxRec
- o dns nameServer
- o dns naptrRec
- o dns nsRec
- o dns nsecRec
- o dns parameter
- o dns policy
- o dns policy64
- o dns policylabel
- o dns proxyRecords
- o dns ptrRec
- o dns records
- o dns soaRec
- o dns srvRec
- o dns stats
- o dns suffix
- o dns txtRec
- o dns view
- o dns zone

dns

The following operations can be performed on "dns":

stat dns

Displays DNS statistics.

Synopsys

```
stat dns [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

Dns queries (Q)

Total number of DNS queries received.

Multi queries (MtQ)

Total number of Multi Query request received.

Dns responses (Rsp)

Total number of DNS responses received.

Server responses (SvrRsp)

Total number of Server responses received.

Total Record updates (TotRecUp)

Total number of record updates.

Auth answers (AuthAns)

Number of queries which were authoritatively answered.

Server queries (SvrQ)

Total number of Server queries sent.

Cache flush called (CaFsh)

Total number of times cache was flushed.

Cache entries flushed (CaEntFsh)

Total number of cache entries flushed.

Non-authoritative entries (PxyEnt)

Total number of non-authoritative entries.

Authoritative entries (AthEnt)

Total number of authoritative entries.

Nonexistent domain (NoDomain)

Number of queries for which no record was found.

Response class unsupported (RspClsEr)

Total number of responses for which response types were unsupported.

Invalid query format (InQFmt)

Total number of queries whose format was invalid.

Stray answers (StryRsp)

Total number of stray answers.

Incorrect RD length (BadRDlen)

Number of DNS responses received with invalid resource data length.

Requests refused (ReqRefused)

Number of DNS requests refused.

NULL Attack (NullAttack)

Total number of queries received where all the counts are 0.

Response type unsupported (RspNoSup)

Total number of responses for which response type requested was unsupported.

Query class unsupported (QClsEr)

Total number of queries for which query class was unsupported.

Invalid response format (InRspFmt)

Total number of responses for which there was a format error.

No answer responses (NoAnswer)

Number of DNS responses received without answer.

Multi queries disabled (MtQErr)

Total number of times a multi query was disabled and received a multi query.

Other errors (OtherErr)

Total number of other errors.

DNS64 queries

Total number of DNS64 queries recieved.

DNS64 answers

Total number of DNS64 answers served.

DNS64 rewrite answers

Total number of DNS64 answers served after rewriting the response.

DNS64 responses

Total number of responses recieved from backend in DNS64 context.

DNS64 GSLB Queries

Total number of DNS64 queries for GSLB domain

DNS64 GSLB Answers

Total number of DNS64 queries served.

DNS64 Total truncated answers

Total number of Answers served with TC bit set in DNS64 context.

DNS64 Total A queries to server

Total number of Queries sent by DNS64 module to backend.

DNS64 Total times AAAA query bypassed

Total number of times AAAA query has been bypassed in DNS64 trnsaction.

DNS64 Total TCP queries

Total number of dns64 queries over TCP

DNS64 Total Active policies

Total number of active dns64 policies

DNS64 Total NODATA Responses

Total number of responses recieved from backend with amount 0

NS queries (NSQ)

Total number of NS queries received.

SOA queries (SOAQ)

Total number of SOA queries received.

PTR queries (PTRQ)

Total number of PTR queries received.

SRV queries (SRVQ)

Total number of SRV queries received.

A responses (ARsp)

Total number of A responses received.

CNAME responses (CNRsp)

Total number of CNAME responses received.

MX responses (MXRsp)

Total number of MX responses received.

ANY responses (ANYRsp)

Total number of ANY responses received.

NS updates (NSUp)

Total number of NS record updates.

SOA updates (SOAUp)

Total number of SOA record updates.

PTR updates (PTRUp)

Total number of PTR record updates.

SRV updates (SRVUp)

Total number of SRV record updates.

AAAA queries (AAAAQ)

Total number of AAAA queries received.

A queries (AQ)

Total number of A queries received.

CNAME queries (CNQ)

Total number of CNAME queries received.

MX queries (MXQ)

Total number of MX queries received.

ANY queries (ANYQ)

Total number of ANY queries received.

AAAA responses (AAAARsp)

Total number of AAAA responses received.

NS responses (NSRsp)

Total number of NS responses received.

SOA responses (SOARsp)

Total number of SOA responses received.

PTR responses (PTRRsp)

Total number of PTR responses received.

SRV responses (SRVRsp)

Total number of SRV responses received.

AAAA updates (AAAAUp)

Total number of AAAA record updates.

A updates (AUp)

Total number of A record updates.

MX updates (MXUp)

Total number of MX record updates.

CNAME updates (CNUp)

Total number of CNAME record updates.

AAAA records (AAAARec)

Total number of AAAA records.

A records (ARec)

Total number of A records.

MX records (MXRec)

Total number of MX records.

CNAME records (CNRec)

Total number of CNAME records.

NS records (NSRec)

Total number of NS records.

SOA records (SOARec)

Total number of SOA records.

PTR records (PTRRec)

Total number of PTR records.

SRV records (SRVRec)

Total number of SRV records.

No AAAA records (NoAAAARec)

Total number of times AAAA record lookup failed.

No A records (NoARec)

Total number of times A record lookup failed.

No MX records (NoMXRec)

Total number of times MX record lookup failed.

No PTR records (NoPTRRec)

Total number of times PTR record lookup failed.

No NS records (NoNSRec)

Total number of times NS record lookup failed.

No CNAME records (NoCNRec)

Total number of times CNAME record lookup failed.

No SOA records (NoSOARec)

Total number of times SOA record lookup failed.

No SRV records (NoSRVRec)

Total number of times SRV record lookup failed.

No ANY records (NoANYrec)

Total number of times ANY query lookup failed.

Unsupported queries (NotSupQ)

Total number of requests for which query type requested was unsupported.

dns aaaaRec

The following operations can be performed on "dns aaaaRec":

[add](#) | [rm](#) | [show](#)

add dns aaaaRec

Creates a AAAA address record for the specified domain name. You cannot modify a AAAA address record.

Synopsys

```
add dns aaaaRec <hostName> <IPv6Address> ... [-TTL <secs>]
```

Arguments

hostName

Domain name.

IPv6Address

One or more IPv6 addresses to assign to the domain name.

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

Default value: 3600

Maximum value: 2147483647

Example

```
add dns aaaarec www.mynw.com 3::4:5 -ttl 10
```

rm dns aaaaRec

Removes an IPv6 address from a AAAA address record. The associated domain name must be specified. If no IPv6 address is specified, all AAAA records that belong to the specified domain name are removed.

Synopsys

```
rm dns aaaaRec <hostName> [<IPv6Address> ...]
```

Arguments

hostName

Domain name.

IPv6Address

IPv6 address(es) of the AAAA record(s) to remove from the specified domain name.

Example

```
rm dns aaaarec www.mynw.com
```

show dns aaaaRec

Displays the AAAA (IPv6) address record for the specified host name. If a hostname is not specified, all configured AAAA records are shown.

Synopsys

```
show dns aaaaRec [<hostName> | -type <type>] [<IPv6Address>]
```

Arguments

hostName

Domain name.

IPv6Address

One or more IPv6 addresses to assign to the domain name.

type

Type of records to display. Available settings function as follows:

- * ADNS - Display all authoritative address records.
- * PROXY - Display all proxy address records.
- * ALL - Display all address records.

Possible values: ALL, ADNS, PROXY

Outputs

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

vServerName

Virtual server name.

authType

Authentication type.

devno

count

stateflag

dns action

The following operations can be performed on "dns action":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add dns action

Add a dns action.

Synopsys

```
add dns action <actionName> <actionType> [-IPAddress <ip_addr|ipv6_addr> ... | -viewName <string> | -preferredLocList <string> ...] [-TTL <secs>]
```

Arguments

actionName

Name of the dns action.

actionType

The type of DNS action that is being configured.

Possible values: ViewName, GslbPrefLoc, noop, Drop, Cache_Bypass, Rewrite_Response

IPAddress

List of IP address to be returned in case of rewrite_response actiontype. They can be of IPV4 or IPV6 type.

In case of set command We will remove all the IP address previously present in the action and will add new once given in set dns action command.

TTL

Time to live, in seconds.

Default value: 3600

Maximum value: 2147483647

viewName

The view name that must be used for the given action.

preferredLocList

The location list in priority order used for the given action.

Example

```
add dns action <actionName> <actionType> (-IPAddress <ip_addr|ipv6_addr> ... | -viewName .
```

rm dns action

Removes a dns Action.

Synopsys

```
rm dns action <actionName>
```

Arguments

actionName

Name of the dns action.

Example

```
rm dns action action1
```

set dns action

Set a dns Action. Use this command to set the values for Ip address and TTL, If Ipaddress is given in set dns action command we will discard the previous set and will apply this new set of ipaddress given.

Synopsys

```
set dns action <actionName> [-IPAddress <ip_addr|ipv6_addr> ...] [-TTL <secs>] [-viewName <string>] [-preferredLocList <string> ...]
```

Arguments

actionName

Name of the dns action.

IPAddress

List of IP address to be returned in case of rewrite_response actiontype. They can be of IPV4 or IPV6 type.

In case of set command We will remove all the IP address previously present in the action and will add new once given in set dns action command.

TTL

Time to live, in seconds.

Default value: 3600

Maximum value: 2147483647

viewName

The view name that must be used for the given action.

preferredLocList

The location list in priority order used for the given action.

Example

```
set dns action <actionName> [-IPAddress <ip_addr|ipv6_addr> ...] [-TTL <secs>] [-viewName
```

unset dns action

Use this command to remove dns action settings.Refer to the set dns action command for meanings of the arguments.

Synopsys

```
unset dns action <actionName> -TTL
```

show dns action

Used to display the action-related information.

Synopsys

```
show dns action [<actionName>]
```

Arguments

actionName

Name of the dns action.

Outputs

actionType

The type of DNS action that is being configured.

TTL

Time to live, in seconds.

IPAddress

List of IP address to be returned in case of rewrite_response actiontype. They can be of IPV4 or IPV6 type.

In case of set command We will remove all the IP address previously present in the action and will add new once given in set dns action command.

viewName

The view name that must be used for the given action.

preferredLocList

The location list in priority order used for the given action.

drop

The dns packet must be dropped.

cacheBypass

By pass dns cache for this.

builtin

Flag to determine whether DNS action is default or not

devno

count

stateflag

Example

```
show dns action <Action-Name> show dns action action1 show dns action
```

dns action64

The following operations can be performed on "dns action64":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add dns action64

Add a dns64 action.

Synopsys

```
add dns action64 <actionName> -Prefix <ipv6_addr|*> [-mappedRule <expression>] [-excludeRule <expression>]
```

Arguments

actionName

Name of the dns64 action.

Prefix

The dns64 prefix to be used if the after evaluating the rules

mappedRule

The expression to select the criteria for ipv4 addresses to be used for synthesis.

Only if the mappedrule is evaluated to true the corresponding ipv4 address is used for synthesis using respective prefix,

otherwise the A RR is discarded

excludeRule

The expression to select the criteria for eliminating the corresponding ipv6 addresses from the response.

Example

```
add dns dns64action <actionName> -prefix f23d:f43e::0/32 [-mappedRule <expr>] [-excludeRu.
```

rm dns action64

Removes a dns64 Action.

Synopsys

```
rm dns action64 <actionName>
```

Arguments

actionName

Name of the dns64 action.

Example

```
rm dns dns64action action1
```

set dns action64

Set a DNS64 Action

Synopsys

```
set dns action64 <actionName> [-Prefix <ipv6_addr|*>] [-mappedRule <expression>] [-excludeRule <expression>]
```

Arguments

actionName

Name of the dns64 action.

Prefix

The dns64 prefix to be used if the after evaluating the rules

mappedRule

The expression to select the criteria for ipv4 addresses to be used for synthesis.

Only if the mappedrule is evaluated to true the corresponding ipv4 address is used for synthesis using respective prefix,

otherwise the A RR is discarded

excludeRule

The expression to select the criteria for eliminating the corresponding ipv6 addresses from the response.

Example

```
set dns dns64action -prefix -mappedrule -excluderule
```

unset dns action64

Use this command to remove dns action64 settings.Refer to the set dns action64 command for meanings of the arguments.

Synopsys

```
unset dns action64 <actionName> [-Prefix] [-mappedRule] [-excludeRule]
```

show dns action64

Used to display the action-related information.

Synopsys

```
show dns action64 [<actionName>]
```

Arguments

actionName

Name of the dns64 action.

Outputs

Prefix

The dns64 prefix to be used if the after evaluating the rules

mappedRule

The expression to select the criteria for ipv4 addresses to be used for synthesis.

Only if the mappedrule is evaluated to true the corresponding ipv4 address is used for synthesis using respective prefix,

otherwise the A RR is discarded

excludeRule

The expression to select the criteria for eliminating the corresponding ipv6 addresses from the response.

builtin

Flag to determine whether dna64action is default or not

devno

count

stateflag

Example

```
show dns dns64action
```

dns addRec

The following operations can be performed on "dns addRec":

[add](#) | [rm](#) | [show](#)

add dns addRec

Creates an IPv4 address record for the specified domain name. You cannot modify an address resource record.

Synopsys

```
add dns addRec <hostName> <IPAddress> ... [-TTL <secs>]
```

Arguments

hostName

Domain name.

IPAddress

One or more IPv4 addresses to assign to the domain name.

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

Default value: 3600

Maximum value: 2147483647

Example

```
Add dns addrec www.mynw.com 65.200.211.139 -ttl 10
```

rm dns addRec

Removes an IPv4 address from an address record. The associated domain name must be specified. If no IPv4 address is specified, all records that belong to the specified domain name are removed.

Synopsys

```
rm dns addRec <hostName> [<IPAddress> ...]
```

Arguments

hostName

Domain name.

IPAddress

IPv4 address(es) of the address records to remove from the specified domain name.

Example

```
rm dns addrec www.mynw.com
```

show dns addRec

Displays the IPv4 address record for the specified host name. If a hostname is not specified, all configured address records are shown.

Synopsys

show dns addRec [<hostName> | -type <type>]

Arguments

hostName

Domain name.

type

The address record type. The type can take 3 values:

ADNS - If this is specified, all of the authoritative address records will be displayed.

PROXY - If this is specified, all of the proxy address records will be displayed.

ALL - If this is specified, all of the address records will be displayed.

Possible values: ALL, ADNS, PROXY

Outputs

IPAddress

IP addresses for the domain name.

TTL

The time to live, in seconds.

vServerName

Virtual server name.

authType

Authentication type.

devno

count

stateflag

dns cnameRec

The following operations can be performed on "dns cnameRec":

[add](#) | [rm](#) | [show](#)

add dns cnameRec

Creates a canonical name (CNAME) record, or alias, for the specified domain name.

Synopsis

```
add dns cnameRec <aliasName> <canonicalName> [-TTL <secs>]
```

Arguments

aliasName

Alias for the canonical domain name.

canonicalName

Canonical domain name.

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

Default value: 3600

Maximum value: 2147483647

Example

```
add dns cnameRec www.mynw.org www.mynw.com -ttl 20
```

rm dns cnameRec

Removes a canonical name (CNAME) record.

Synopsis

```
rm dns cnameRec <aliasName>
```

Arguments

aliasName

Alias for which to remove the CNAME record.

Example

```
rm dns cnamerec www.mynw.org
```

show dns cnameRec

Displays the canonical name (CNAME) records configured for the specified alias. If no alias is specified, all configured CNAME records are displayed

Synopsys

```
show dns cnameRec [<aliasName> | -type <type>]
```

Arguments

aliasName

Alias for which to display CNAME records.

type

Type of records to display. Available settings function as follows:

- * ADNS - Display all authoritative address records.
- * PROXY - Display all proxy address records.
- * ALL - Display all address records.

Possible values: ALL, ADNS, PROXY

Default value: ADNS

Outputs

canonicalName

Canonical domain name.

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

vServerName

GSLB Virtual server name to which this domain is bound

authType

Record type.

devno

count

stateflag

Example

```
show dns cnameRec www.mynw.org
```

dns global

The following operations can be performed on "dns global":

[bind](#) | [unbind](#) | [show](#)

bind dns global

Binds the specified DNS policy globally.

Synopsys

```
bind dns global <policyName> <priority> [-gotoPriorityExpression <string>] [-type <type>] [-invoke (<labelType> <labelName>)]
```

Arguments

policyName

Name of the DNS policy to bind globally.

priority

Integer specifying the policy's priority. The lower the number, the higher the priority.

Minimum value: 1

gotoPriorityExpression

Expression or other value specifying the next policy to be evaluated if the current policy evaluates to TRUE. Specify one of the following values:

- * NEXT - Evaluate the policy with the next higher priority number.
- * END - End policy evaluation.
- * USE_INVOCATION_RESULT - Applicable if this policy invokes another policy label. If the final goto in the invoked policy label has a value of END, the evaluation stops. If the final goto is anything other than END, the current policy label performs a NEXT.
- * A default syntax expression that evaluates to a number.

If you specify an expression, the number to which it evaluates determines the next policy to evaluate, as follows:

- * If the expression evaluates to a higher numbered priority, the policy with that priority is evaluated next.
- * If the expression evaluates to the priority of the current policy, the policy with the next higher numbered priority is evaluated next.
- * If the expression evaluates to a priority number that is numerically higher than the highest numbered priority, policy evaluation ends.

An UNDEF event is triggered if:

- * The expression is invalid.
- * The expression evaluates to a priority number that is numerically lower than the current policy's priority.
- * The expression evaluates to a priority number that is between the current policy's priority number (say, 30) and the highest priority number (say, 100), but does not match any configured priority number (for example, the expression evaluates to the number 85). This example assumes that the priority number increments by 10 for every successive policy, and therefore a priority number of 85 does not exist in the policy label.

type

Type of global bind point to which to bind the DNS policy.

Possible values: REQ_OVERRIDE, REQ_DEFAULT, RES_OVERRIDE, RES_DEFAULT

invoke

Invoke flag.

labelType

Type of policy label invocation.

Possible values: policylabel

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

Example

```
bind dns global pol9 9
```

unbind dns global

Unbinds the specified DNS policy from the global bind point.

Synopsys

```
unbind dns global <policyName> [-type <type>]
```

Arguments

policyName

Name of the DNS policy to unbind.

type

Type of global bind point to which to bind the DNS policy.

Possible values: REQ_OVERRIDE, REQ_DEFAULT, RES_OVERRIDE, RES_DEFAULT

Example

```
unbind dns global pol9
```

show dns global

Displays the DNS policies bound to the specified global bind point. If a global bind point is not specified, the command displays the global bind points that have policies bound to them, and the number of policies bound to each of those bind points.

Synopsys

```
show dns global [-type <type>]
```

Arguments

type

Type of global bind point for which to show bound policies.

Possible values: REQ_OVERRIDE, REQ_DEFAULT, RES_OVERRIDE, RES_DEFAULT

Outputs

stateflag

policyName

Name of the dns policy.

priority

Specifies the priority of the policy with which it is bound. Maximum allowed priority should be less than 65535

gotoPriorityExpression

Expression or other value specifying the next policy to be evaluated if the current policy evaluates to TRUE. Specify one of the following values:

- * NEXT - Evaluate the policy with the next higher priority number.
- * END - End policy evaluation.
- * USE_INVOCATION_RESULT - Applicable if this policy invokes another policy label. If the final goto in the invoked policy label has a value of END, the evaluation stops. If the final goto is anything other than END, the current policy label performs a NEXT.
- * A default syntax expression that evaluates to a number.

If you specify an expression, the number to which it evaluates determines the next policy to evaluate, as follows:

- * If the expression evaluates to a higher numbered priority, the policy with that priority is evaluated next.
- * If the expression evaluates to the priority of the current policy, the policy with the next higher numbered priority is evaluated next.
- * If the expression evaluates to a priority number that is numerically higher than the highest numbered priority, policy evaluation ends.

An UNDEF event is triggered if:

- * The expression is invalid.
- * The expression evaluates to a priority number that is numerically lower than the current policy's priority.
- * The expression evaluates to a priority number that is between the current policy's priority number (say, 30) and the highest priority number (say, 100), but does not match any configured priority number (for example, the expression evaluates to the number 85). This example assumes that the priority number increments by 10 for every successive policy, and therefore a priority number of 85 does not exist in the policy label.

invoke

Invoke flag.

labelType

Type of policy label invocation.

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

numpol

The number of policies bound to the bindpoint.

flowType

flowtype of the bound rewrite policy.

flags**upgraded**

It is internally used to tell that the policy is a upgraded policy.

builtin

Flag to determine whether DNS policy binding is default or not

devno**count**

Example

```
show dns global show dns global -type REQ_DEFAULT show dns global -type RES_DEFAULT
```

dns key

The following operations can be performed on "dns key":

[add](#) | [create](#) | [set](#) | [unset](#) | [rm](#) | [show](#)

add dns key

Adds a DNS key to the zone that is specified in the key file.

Synopsis

```
add dns key <keyName> <publickey> <privatekey> [-expires <positive_integer> [<units>]] [-notificationPeriod <positive_integer> [<units>]] [-TTL <secs>]
```

Arguments

keyName

Name of the public-private key pair to publish in the zone.

publickey

File name of the public key.

privatekey

File name of the private key.

expires

Time period for which to consider the key valid, after the key is used to sign a zone.

Default value: 120

Minimum value: 1

Maximum value: 32767

units

Units for the notification period.

Possible values: MINUTES, HOURS, DAYS

Default value: DAYS

notificationPeriod

Time at which to generate notification of key expiration, specified as number of days, hours, or minutes before expiry. Must be less than the expiry period. The notification is an SNMP trap sent to an SNMP manager. To enable the appliance to send the trap, enable the DNSKEY-EXPIRY SNMP alarm.

Default value: 7

Minimum value: 1

Maximum value: 32767

TTL

Time to Live (TTL), in seconds, for the DNSKEY resource record created in the zone. TTL is the time for which the record must be cached by the DNS proxies. If the TTL is not specified, either the DNS zone's minimum TTL or the default value of 3600 is used.

Default value: 3600

Maximum value: 2147483647

Example

```
add dns key secure.example.zsk -public secure.example-rsasha1-1024.key -private /nsconfig/dns/secure.example.zsk.private
```

create dns key

Creates a public-private key pair to use for signing a DNS zone. The keys are created in the /nsconfig/dns/ directory on the NetScaler appliance. The private, public, and DS key files are created with names having the format <prefix>.<key/private/ds>.

Synopsis

```
create dns key -zoneName <string> -keyType <keyType> -algorithm RSASHA1 -keySize <positive_integer> -
fileNamePrefix <string>
```

Arguments

zoneName

Name of the zone for which to create a key.

keyType

Type of key to create.

Possible values: KSK, KeySigningKey, ZSK, ZoneSigningKey

Default value: NS_DNSKEY_ZSK

algorithm

Algorithm to generate for zone signing.

Possible values: RSASHA1

Default value: NS_DNSKEYALGO_RSASHA1

keySize

Size of the key, in bits.

Default value: 512

Minimum value: 0

fileNamePrefix

Common prefix for the names of the generated public and private key files and the Delegation Signer (DS) resource record. During key generation, the .key, .private, and .ds suffixes are appended automatically to the file name prefix to produce the names of the public key, the private key, and the DS record, respectively.

Example

```
create dns key -zone dnssec.bar -algorithm RSASHA1 -keySize 1024
```

set dns key

Modifies the specified parameters of a DNS key. Note: If you change the expiry time period of a key, the NetScaler appliance, using the modified key, automatically re-signs all the resource records in the zone, provided that the zone is currently signed with the particular key.

Synopsis

```
set dns key <keyName> [-expires <positive_integer> [<units>]] [-notificationPeriod <positive_integer> [<units>]] [-
TTL <secs>]
```


Arguments

keyName

Name of the public-private key pair.

expires

Time period for which to consider the key valid, after the key is used to sign a zone.

Default value: 120

Minimum value: 1

Maximum value: 32767

units

Units for the notification period.

Possible values: MINUTES, HOURS, DAYS

Default value: DAYS

notificationPeriod

Time at which to generate notification of key expiration, specified as number of days, hours, or minutes before expiry. Must be less than the expiry period. The notification is an SNMP trap sent to an SNMP manager. To enable the appliance to send the trap, enable the DNSKEY-EXPIRY SNMP alarm.

Default value: 7

Minimum value: 1

Maximum value: 32767

TTL

Time to Live (TTL), in seconds, for the DNSKEY resource record created in the zone. TTL is the time for which the record must be cached by the DNS proxies. If the TTL is not specified, either the DNS zone's minimum TTL or the default value of 3600 is used.

Default value: 3600

Maximum value: 2147483647

Example

```
add dns key secure.example.zsk -public secure.example-rsasha1-1024.key -private /nsconfi
```

unset dns key

Use this command to remove dns key settings. Refer to the set dns key command for meanings of the arguments.

Synopsis

```
unset dns key <keyName> [-expires] [-units] [-notificationPeriod] [-units] [-TTL]
```

rm dns key

Removes a DNS key.

Synopsis

```
rm dns key <keyName>
```

Arguments

keyName

Name of the public-private key pair.

Example

```
rm dns key secure.example.zsk
```

show dns key

Displays the parameters of the specified DNS key. If no DNS key name is specified, all configured DNS keys are shown. Note: You cannot view the parameters of a public/private key file. You can view the parameters of a key after you have published it in a DNS zone by using either the add dns key command or the DNS > Zones > Sign/Unsign DNS Zone dialog box.

Synopsys

```
show dns key [<keyName>]
```

Arguments

keyName

Name of the public-private key pair.

Outputs

publickey

File name of the public key.

privatekey

File name of the private key.

expires

Number of days since signing with this key, when the key expires.

units

Units for the notification period.

notificationPeriod

Time at which to generate notification of key expiration, specified as number of days, hours, or minutes before expiry. Must be less than the expiry period. The notification is an SNMP trap sent to an SNMP manager. To enable the appliance to send the trap, enable the DNSKEY-EXPIRY SNMP alarm.

TTL

Time to Live (TTL), in seconds, for the DNSKEY resource record created in the zone. TTL is the time for which the record must be cached by the DNS proxies. If the TTL is not specified, either the DNS zone's minimum TTL or the default value of 3600 is used.

zoneName

Name of the zone for which the key is created.

devno

count

stateflag

Example

```
show dns key
```

dns mxRec

The following operations can be performed on "dns mxRec":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add dns mxRec

Creates a mail exchange (MX) record for the specified domain name.

Synopsys

```
add dns mxRec <domain> -mx <string> -pref <positive_integer> [-TTL <secs>]
```

Arguments

domain

Domain name for which to add the MX record.

mx

Host name of the mail exchange server.

pref

Priority number to assign to the mail exchange server. A domain name can have multiple mail servers, with a priority number assigned to each server. The lower the priority number, the higher the mail server's priority. When other mail servers have to deliver mail to the specified domain, they begin with the mail server with the lowest priority number, and use other configured mail servers, in priority order, as backups.

Minimum value: 0

Maximum value: 65535

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

Default value: 3600

Maximum value: 2147483647

rm dns mxRec

Removes the specified mail exchange (MX) record from the specified domain.

Synopsys

```
rm dns mxRec <domain> <mx>
```

Arguments

domain

Domain name.

mx

Host name of the mail exchange server.

set dns mxRec

Modifies the priority number and TTL of the mail exchange (MX) record.

Synopsys

```
set dns mxRec <domain> -mx <string> [-pref <positive_integer>] [-TTL <secs>]
```

Arguments

domain

Domain of the MX record to be modified.

mx

Host name of the mail exchange server to be modified.

pref

Priority number to assign to the mail exchange server. A domain name can have multiple mail servers, with a priority number assigned to each server. The lower the priority number, the higher the mail server's priority. When other mail servers have to deliver mail to the specified domain, they begin with the mail server with the lowest priority number, and use other configured mail servers, in priority order, as backups.

Minimum value: 0

Maximum value: 65535

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

Default value: 3600

Maximum value: 2147483647

unset dns mxRec

Use this command to remove dns mxRec settings. Refer to the set dns mxRec command for meanings of the arguments.

Synopsys

```
unset dns mxRec <domain> -mx <string> -TTL
```

show dns mxRec

Displays the mail exchange (MX) records for the specified domain. If no domain name is specified, all configured mail exchange records are shown.

Synopsys

```
show dns mxRec [<domain> | -type <type>]
```

Arguments

domain

Domain name.

type

Type of records to display. Available settings function as follows:

- * ADNS - Display all authoritative address records.
- * PROXY - Display all proxy address records.
- * ALL - Display all address records.

Possible values: ALL, ADNS, PROXY

Default value: ADNS

Outputs

mx

Host name of the mail exchange server.

pref

Priority number to assign to the mail exchange server. A domain name can have multiple mail servers, with a priority number assigned to each server. The lower the priority number, the higher the mail server's priority. When other mail servers have to deliver mail to the specified domain, they begin with the mail server with the lowest priority number, and use other configured mail servers, in priority order, as backups.

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

authType

Record type.

devno**count****stateflag**

dns nameServer

The following operations can be performed on "dns nameServer":

[add](#) | [rm](#) | [enable](#) | [disable](#) | [show](#)

add dns nameServer

Adds a name server to the appliance. Following are the two types of name servers that can be added: * IP address-based name server - An external name server to contact for domain name resolution. If multiple IP address-based name servers are configured on the appliance, and the local parameter is not set on any of them, incoming DNS queries are load balanced across all the name servers, in round robin fashion. * Virtual server-based name server - A DNS virtual server configured in the NetScaler appliance. If you want more fine-grained control on how external DNS name servers are load balanced (for example, you want a load balancing method other than round robin), you configure a DNS virtual server on the appliance, bind the external name servers as its services, and then specify the name of the virtual server in this command.

Synopsys

```
add dns nameServer ((<IP> [-local]) | <dnsVserverName>) [-state ( ENABLED | DISABLED )] [-type <type>]
```

Arguments

IP

IP address of an external name server or, if the Local parameter is set, IP address of a local DNS server (LDNS).

dnsVserverName

Name of a DNS virtual server. Overrides any IP address-based name servers configured on the NetScaler appliance.

local

Mark the IP address as one that belongs to a local recursive DNS server on the NetScaler appliance. The appliance recursively resolves queries received on an IP address that is marked as being local. For recursive resolution to work, the global DNS parameter, Recursion, must also be set.

If no name server is marked as being local, the appliance functions as a stub resolver and load balances the name servers.

state

Administrative state of the name server.

Possible values: ENABLED, DISABLED

Default value: ENABLED

type

Protocol used by the name server. UDP_TCP is not valid if the name server is a DNS virtual server configured on the appliance.

Possible values: UDP, TCP, UDP_TCP

Default value: UDP

Example

Adding an-IP based nameserver IP: `add nameserver 10.102.4.1`, Adding a vservice-based name

rm dns nameServer

Removes a name server from the NetScaler appliance. If the name server is an IP-address based external name server, the name server entry is removed. If the name server is a DNS virtual server on the appliance, the virtual server is not removed, but it is no longer used to resolve domain names.

Synopsys

```
rm dns nameServer (<IP> | <dnsVserverName>)
```

Arguments

IP

IP address of the name server.

dnsVserverName

Name of the DNS virtual server.

Example

```
Deleting an IP-based nameserver:    rm nameserver 10.102.4.1,  Deleting a vserver-based nar
```

enable dns nameServer

Enables a name server.

Synopsys

```
enable dns nameServer (<IP> | <dnsVserverName>)
```

Arguments

IP

IP address of the name server.

dnsVserverName

Name of the DNS virtual server.

Example

```
enable dns nameserver 10.14.43.149
```

disable dns nameServer

Disables a name server.

Synopsys

```
disable dns nameServer (<IP> | <dnsVserverName>)
```

Arguments

IP

IP address of the name server.

dnsVserverName

Name of the DNS virtual server.

Example

```
disable dns nameserver 10.14.43.149
```

show dns nameServer

Displays the name servers configured on the NetScaler appliance, along with their administrative states.

Synopsys

```
show dns nameServer [<IP> | <dnsVserverName>]
```

Arguments

IP

IP address of the name server.

dnsVserverName

Name of the DNS virtual server.

Outputs

serviceName

The name of the dns vserver.

port

Port of the service.

type

Protocol used by the name server. UDP_TCP is not valid if the name server is a DNS virtual server configured on the appliance.

state

Administrative state of the name server.

nameserverstate

State of the server.

local

ip is a local recursive nameserver.

adminState

CIMonOwner

Tells the mon owner of the service.

CIMonView

Tells the view id by which state of the service is updated.

devno

count

stateflag

dns naptrRec

The following operations can be performed on "dns naptrRec":

[add](#) | [rm](#) | [show](#)

add dns naptrRec

Creates an NAPTR record. Each resource record is stored with a unique, internally generated record ID, which you can view and use to delete the record.

Synopsys

```
add dns naptrRec <domain> <order> <preference> [-flags <string>] [-services <string>] (-regexp <expression> | -replacement <string>) [-TTL <secs>]
```

Arguments

domain

Name of the domain for the NAPTR record.

order

An integer specifying the order in which the NAPTR records MUST be processed in order to accurately represent the ordered list of Rules. The ordering is from lowest to highest

Minimum value: 0

Maximum value: 65535

preference

An integer specifying the preference of this NAPTR among NAPTR records having same order. lower the number, higher the preference.

Minimum value: 0

Maximum value: 65535

flags

flags for this NAPTR.

services

Service Parameters applicable to this delegation path.

regexp

The regular expression, that specifies the substitution expression for this NAPTR

replacement

The replacement domain name for this NAPTR.

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

Default value: 3600

Maximum value: 2147483647

Example

TBD

rm dns naptrRec

Removes the specified NAPTR record from the specified domain.

Synopsys

```
rm dns naptrRec <domain> ((<order> <preference> [-flags <string>] [-services <string>] (-regexp <expression> | -replacement <string>)) | -recordId <positive_integer>@)
```

Arguments

domain

Name of the domain for the NAPTR record.

order

An integer specifying the order in which the NAPTR records MUST be processed in order to accurately represent the ordered list of Rules. The ordering is from lowest to highest

Minimum value: 0

Maximum value: 65535

recordId

Unique, internally generated record ID. View the details of the naptr record to obtain its record ID. Records can be removed by either specifying the domain name and record id OR by specifying

domain name and all other naptr record attributes as was supplied during the add command.

Minimum value: 1

Maximum value: 65535

preference

An integer specifying the preference of this NAPTR among NAPTR records having same order. lower the number, higher the preference.

Minimum value: 0

Maximum value: 65535

flags

flags for this NAPTR.

services

Service Parameters applicable to this delegation path.

regexp

The regular expression, that specifies the substitution expression for this NAPTR

replacement

The replacement domain name for this NAPTR.

Example

TBD

show dns naptrRec

Displays NAPTR records owned by the specified domain. If no domain name is specified, all configured NAPTR records are shown.

Synopsys

```
show dns naptrRec [<domain> | -type <type>]
```

Arguments

domain

Name of the domain for the NAPTR record.

type

Type of records to display. Available settings function as follows:

* ADNS - Display all authoritative address records.

* PROXY - Display all proxy address records.

* ALL - Display all address records.

Possible values: ALL, ADNS, PROXY

Default value: ADNS

Outputs

order

An integer specifying the order in which the NAPTR records MUST be processed in order to accurately represent the ordered list of Rules. The ordering is from lowest to highest

preference

An integer specifying the preference of this NAPTR among NAPTR records having same order. lower the number, higher the preference.

flags

flags for this NAPTR.

services

Service Parameters applicable to this delegation path.

regexp

The regular expression, that specifies the substitution expression for this NAPTR

replacement

The replacement domain name for this NAPTR.

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

recordId

authType

Authentication type.

devno**count****stateflag**

Example

```
show dns naptrRec spf.m.test. show dns naptrRec
```

dns nsRec

The following operations can be performed on "dns nsRec":

[add](#) | [rm](#) | [show](#)

add dns nsRec

Creates a name server record for the specified domain.

Synopsys

```
add dns nsRec <domain> <nameServer> [-TTL <secs>]
```

Arguments

domain

Domain name.

nameServer

Host name of the name server to add to the domain.

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

Default value: 3600

Maximum value: 2147483647

rm dns nsRec

Removes the specified name server record from the specified domain.

Synopsys

```
rm dns nsRec <domain> <nameServer>
```

Arguments

domain

Domain name.

nameServer

Name server to remove.

show dns nsRec

Displays the name server records for the specified domain. If no domain name is specified, all configured name server records are shown.

Synopsys

```
show dns nsRec [<domain> | -type <type>]
```

Arguments

domain

Domain name.

type

Type of records to display. Available settings function as follows:

- * ADNS - Display all authoritative address records.
- * PROXY - Display all proxy address records.
- * ALL - Display all address records.

Possible values: ALL, ADNS, PROXY

Outputs

nameServer

Host name of the name server to add to the domain.

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

authType

Record type.

devno

count

stateflag

dns nsecRec

The following operations can be performed on "dns nsecRec":

show dns nsecRec

Displays the NextSECure (NSEC) resource records created for the specified domain name.

Synopsys

```
show dns nsecRec [<hostName> | -type <type>]
```

Arguments

hostName

Name of the domain.

type

Type of records to display. Available settings function as follows:

- * ADNS - Display all authoritative address records.
- * PROXY - Display all proxy address records.
- * ALL - Display all address records.

Possible values: ALL, ADNS, PROXY

Outputs

nextNsec

Next nsec record in the chain

nsecBitarray

Bit array representing the different record types configured for the domain name

nextRecs

An array of record types associated with the nsec record.

TTL

Time to Live (TTL), in seconds, for the record.

devno

count

stateflag

Example

```
show dns nsecRec foo.bar
```


dns parameter

The following operations can be performed on "dns parameter":

[set](#) | [unset](#) | [show](#)

set dns parameter

Modifies global DNS parameters on the NetScaler appliance.

Synopsys

```
set dns parameter [-retries <positive_integer>] [-minTTL <secs>] [-maxTTL <secs>] [-cacheRecords ( YES | NO )] [-nameLookupPriority ( WINS | DNS )] [-recursion ( ENABLED | DISABLED )] [-resolutionOrder <resolutionOrder>] [-dnssec ( ENABLED | DISABLED )] [-maxPipeline <positive_integer>] [-dnsRootReferral ( ENABLED | DISABLED )] [-dns64Timeout <msecs>]
```

Arguments

retries

Maximum number of retry attempts when no response is received for a query sent to a name server. Applies to end resolver and forwarder configurations.

Default value: 5

Minimum value: 1

Maximum value: 5

minTTL

Minimum permissible time to live (TTL) for all records cached in the DNS cache by DNS proxy, end resolver, and forwarder configurations. If the TTL of a record that is to be cached is lower than the value configured for minTTL, the TTL of the record is set to the value of minTTL before caching. When you modify this setting, the new value is applied only to those records that are cached after the modification. The TTL values of existing records are not changed.

Maximum value: 604800

maxTTL

Maximum time to live (TTL) for all records cached in the DNS cache by DNS proxy, end resolver, and forwarder configurations. If the TTL of a record that is to be cached is higher than the value configured for maxTTL, the TTL of the record is set to the value of maxTTL before caching. When you modify this setting, the new value is applied only to those records that are cached after the modification. The TTL values of existing records are not changed.

Default value: 604800

Minimum value: 1

Maximum value: 604800

cacheRecords

Cache resource records in the DNS cache. Applies to resource records obtained through proxy configurations only. End resolver and forwarder configurations always cache records in the DNS cache, and you cannot disable this behavior. When you disable record caching, the appliance stops caching server responses. However, cached records are not flushed. The appliance does not serve requests from the cache until record caching is enabled again.

Possible values: YES, NO

Default value: YES

nameLookupPriority

Type of lookup (DNS or WINS) to attempt first. If the first-priority lookup fails, the second-priority lookup is attempted. Used only by the SSL VPN feature.

Possible values: WINS, DNS

Default value: WINS

recursion

Function as an end resolver and recursively resolve queries for domains that are not hosted on the NetScaler appliance. Also resolve queries recursively when the external name servers configured on the appliance (for a forwarder configuration) are unavailable. When external name servers are unavailable, the appliance queries a root server and resolves the request recursively, as it does for an end resolver configuration.

Possible values: ENABLED, DISABLED

Default value: DISABLED

resolutionOrder

Type of DNS queries (A, AAAA, or both) to generate during the routine functioning of certain NetScaler features, such as SSL VPN, cache redirection, and the integrated cache. The queries are sent to the external name servers that are configured for the forwarder function. If you specify both query types, you can also specify the order. Available settings function as follows:

* OnlyAQuery. Send queries for IPv4 address records (A records) only.

* OnlyAAAAQuery. Send queries for IPv6 address records (AAAA records) instead of queries for IPv4 address records (A records).

* AThenAAAAQuery. Send a query for an A record, and then send a query for an AAAA record if the query for the A record results in a NODATA response from the name server.

* AAAAThenAQuery. Send a query for an AAAA record, and then send a query for an A record if the query for the AAAA record results in a NODATA response from the name server.

Possible values: OnlyAQuery, OnlyAAAAQuery, AThenAAAAQuery, AAAAThenAQuery

Default value: OnlyAQuery

dnssec

Enable or disable the Domain Name System Security Extensions (DNSSEC) feature on the appliance. Note: Even when the DNSSEC feature is enabled, forwarder configurations (used by internal NetScaler features such as SSL VPN and Cache Redirection for name resolution) do not support the DNSSEC OK (DO) bit in the EDNS0 OPT header.

Possible values: ENABLED, DISABLED

Default value: ENABLED

maxPipeline

Maximum number of concurrent DNS requests to allow on a single client connection, which is identified by the <clientip:port>-<vserver ip:port> tuple. A value of 0 (zero) applies no limit to the number of concurrent DNS requests allowed on a single client connection.

Default value: NSNATPCB_MAXPIPELINE

Minimum value: 0

dnsRootReferral

Send a root referral if a client queries a domain name that is unrelated to the domains configured/cached on the NetScaler appliance. If the setting is disabled, the appliance sends a blank response instead of a root referral. Applicable to domains for which the appliance is authoritative. Disable the parameter when the appliance is under attack from a client that is sending a flood of queries for unrelated domains.

Possible values: ENABLED, DISABLED

Default value: DISABLED

dns64Timeout

While doing DNS64 resolution, this parameter specifies the time to wait before sending an A query if no response is received from backend DNS server for AAAA query.

Default value: -1

Maximum value: 10000

unset dns parameter

Use this command to remove dns parameter settings. Refer to the set dns parameter command for meanings of the arguments.

Synopsys

unset dns parameter [-retries] [-minTTL] [-maxTTL] [-cacheRecords] [-nameLookupPriority] [-recursion] [-resolutionOrder] [-dnssec] [-maxPipeline] [-dnsRootReferral] [-dns64Timeout]

show dns parameter

Displays the global DNS parameters.

Synopsys

show dns parameter

Outputs

retries

Maximum number of retry attempts when no response is received for a query sent to a name server. Applies to end resolver and forwarder configurations.

minTTL

Minimum permissible time to live (TTL) for all records cached in the DNS cache by DNS proxy, end resolver, and forwarder configurations. If the TTL of a record that is to be cached is lower than the value configured for minTTL, the TTL of the record is set to the value of minTTL before caching. When you modify this setting, the new value is applied only to those records that are cached after the modification. The TTL values of existing records are not changed.

maxTTL

Maximum time to live (TTL) for all records cached in the DNS cache by DNS proxy, end resolver, and forwarder configurations. If the TTL of a record that is to be cached is higher than the value configured for maxTTL, the TTL of the record is set to the value of maxTTL before caching. When you modify this setting, the new value is applied only to those records that are cached after the modification. The TTL values of existing records are not changed.

nameLookupPriority

Type of lookup (DNS or WINS) to attempt first. If the first-priority lookup fails, the second-priority lookup is attempted. Used only by the SSL VPN feature.

cacheRecords

Cache resource records in the DNS cache. Applies to resource records obtained through proxy configurations only. End resolver and forwarder configurations always cache records in the DNS cache, and you cannot disable this behavior. When you disable record caching, the appliance stops caching server responses. However, cached records are not flushed. The appliance does not serve requests from the cache until record caching is enabled again.

recursion

Function as an end resolver and recursively resolve queries for domains that are not hosted on the NetScaler appliance. Also resolve queries recursively when the external name servers configured on the appliance (for

a forwarder configuration) are unavailable. When external name servers are unavailable, the appliance queries a root server and resolves the request recursively, as it does for an end resolver configuration.

resolutionOrder

Type of DNS queries (A, AAAA, or both) to generate during the routine functioning of certain NetScaler features, such as SSL VPN, cache redirection, and the integrated cache. The queries are sent to the external name servers that are configured for the forwarder function. If you specify both query types, you can also specify the order. Available settings function as follows:

- * OnlyAQuery. Send queries for IPv4 address records (A records) only.

- * OnlyAAAAQuery. Send queries for IPv6 address records (AAAA records) instead of queries for IPv4 address records (A records).

- * AThenAAAAQuery. Send a query for an A record, and then send a query for an AAAA record if the query for the A record results in a NODATA response from the name server.

- * AAAAThenAQuery. Send a query for an AAAA record, and then send a query for an A record if the query for the AAAA record results in a NODATA response from the name server.

dnssec

Enable or disable the Domain Name System Security Extensions (DNSSEC) feature on the appliance. Note: Even when the DNSSEC feature is enabled, forwarder configurations (used by internal NetScaler features such as SSL VPN and Cache Redirection for name resolution) do not support the DNSSEC OK (DO) bit in the EDNS0 OPT header.

maxPipeline

Maximum value of the concurrent DNS pipeline. A setting of zero makes the pipeline infinite

dnsRootReferral

Send a root referral if a client queries a domain name that is unrelated to the domains configured/cached on the NetScaler appliance. If the setting is disabled, the appliance sends a blank response instead of a root referral. Applicable to domains for which the appliance is authoritative. Disable the parameter when the appliance is under attack from a client that is sending a flood of queries for unrelated domains.

dns64Timeout

While doing DNS64 resolution, this parameter specifies the time to wait before sending an A query if no response is received from backend DNS server for AAAA query.

dns policy

The following operations can be performed on "dns policy":

[add](#) | [rm](#) | [set](#) | [show](#)

add dns policy

Creates a DNS policy.

Synopsis

```
add dns policy <name> <rule> <actionName>
```

Arguments

name

Name for the DNS policy.

rule

Expression against which DNS traffic is evaluated. Written in the default syntax.

Note:

- * On the command line interface, if the expression includes blank spaces, the entire expression must be enclosed in double quotation marks.

- * If the expression itself includes double quotation marks, you must escape the quotations by using the character.

- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

Example: CLIENT.UDP.DNS.DOMAIN.EQ("domainname")

actionName

Name of the DNS action to perform when the rule evaluates to TRUE. The built in actions function as follows:

- * dns_default_act_Drop. Drop the DNS request.

- * dns_default_act_Cachebypass. Bypass the DNS cache and forward the request to the name server.

You can create custom actions by using the add dns action command in the CLI or the DNS > Actions > Create DNS Action dialog box in the NetScaler configuration utility.

Example

```
add dns policy poll "dns.req.question.type.ne(aaaa)" -actionName act1 add dns policy pol2
```

rm dns policy

Removes a DNS policy.

Synopsis

```
rm dns policy <name>
```

Arguments

name

Name of the DNS policy to remove.

set dns policy

Modifies the parameters of the specified DNS policy.

Synopsis

```
set dns policy <name> [<rule>] [-actionName <string>]
```

Arguments

name

Name of the DNS policy.

rule

Expression against which DNS traffic is evaluated. Written in the default syntax.

Note:

- * On the command line interface, if the expression includes blank spaces, the entire expression must be enclosed in double quotation marks.

- * If the expression itself includes double quotation marks, you must escape the quotations by using the character.

- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

Example: CLIENT.UDP.DNS.DOMAIN.EQ("domainname")

actionName

Name of the DNS action to perform when the rule evaluates to TRUE. The built in actions function as follows:

- * dns_default_act_Drop. Drop the DNS request.

- * dns_default_act_Cachebypass. Bypass the DNS cache and forward the request to the name server.

You can create custom actions by using the add dns action command in the CLI or the DNS > Actions > Create DNS Action dialog box in the NetScaler configuration utility.

Example

```
set dns policy pol1 -rule "dns.req.question.type.ne(aaaa)" set dns policy pol2 -rule "CLI"
```

show dns policy

Displays the parameters of the specified DNS policy or, if no policy name is specified, all configured DNS policies.

Synopsis

```
show dns policy [<name>]
```

Arguments

name

Name of the DNS policy.

Outputs

rule

The expression to be used by the dns policy.

viewName

The view name that must be used for the given policy

preferredLocation

The location used for the given policy. This is deprecated attribute. Please use -prefLocList

preferredLocList

The location list in priority order used for the given policy.

hits

The number of times the policy has been hit.

undefHits

Number of Undef hits.

drop

The dns packet must be dropped.

actionName

Name of the DNS action to perform when the rule evaluates to TRUE. The built in actions function as follows:

* dns_default_act_Drop. Drop the DNS request.

* dns_default_act_Cachebypass. Bypass the DNS cache and forward the request to the name server.

You can create custom actions by using the add dns action command in the CLI or the DNS > Actions > Create DNS Action dialog box in the NetScaler configuration utility.

cacheBypass

By pass dns cache for this.

activePolicy

Indicates whether policy is bound or not.

boundTo

Location where policy is bound

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

labelType

Type of policy label invocation.

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

description

Description of the policy

builtin

Flag to determine whether DNS policy is default or not

stateflag**devno****count**

dns policy64

The following operations can be performed on "dns policy64":

[add](#) | [rm](#) | [set](#) | [show](#)

add dns policy64

Creates a DNS64 Policy.

Synopsys

```
add dns policy64 <name> -rule <expression> -action <string>
```

Arguments

name

Name for the DNS64 policy.

rule

Expression against which DNS traffic is evaluated. Written in the default syntax.

Note:

- * On the command line interface, if the expression includes blank spaces, the entire expression must be enclosed in double quotation marks.

- * If the expression itself includes double quotation marks, you must escape the quotations by using the character.

- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

Example: CLIENT.IP.SRC.IN_SUBNET(23.34.0.0/16)

action

Name of the DNS64 action to perform when the rule evaluates to TRUE. The built in actions function as follows:

- * A default dns64 action with prefix <default prefix> and mapped and exclude are any

You can create custom actions by using the add dns action command in the CLI or the DNS64 > Actions > Create DNS64 Action dialog box in the NetScaler configuration utility.

Example

```
add dns64 policy poll "client.ip.src.in_subnet(23.43.0.0/16)" -action act1
```

rm dns policy64

Removes a DNS64 Policy.

Synopsys

```
rm dns policy64 <name>
```

Arguments

name

Name of the DNS64 policy to be removed.

set dns policy64

Modifies the parameters of the specified DNS64 policy.

Synopsys

```
set dns policy64 <name> [-rule <expression>] [-action <string>]
```

Arguments

name

Name of the DNS policy.

rule

Expression against which DNS traffic is evaluated. Written in the default syntax.

Note:

- * On the command line interface, if the expression includes blank spaces, the entire expression must be enclosed in double quotation marks.

- * If the expression itself includes double quotation marks, you must escape the quotations by using the character.

- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

Example: CLIENT.IP.SRC.IN_SUBNET(23.34.0.0/16)

action

Name of the DNS64 action to perform when the rule evaluates to TRUE. The built in actions function as follows:

- * A default dns64 action with prefix <default prefix> and mapped and exclude are any

You can create custom actions by using the add dns action command in the CLI or the DNS64 > Actions > Create DNS64 Action dialog box in the NetScaler configuration utility.

Example

```
set dns policy pol2 -rule "CLIENT.IP.SRC.IN_SUBNET(1.1.1.1/24)"
```

show dns policy64

Displays the parameters of the specified DNS64 policy or, if no policy name is specified, all configured DNS64 policies.

Synopsys

```
show dns policy64 [<name>]
```

Arguments

name

Name of the DNS64 policy.

Outputs

rule

The expression to be used by the dns policy.

hits

The number of times the policy has been hit.

action

Name of the DNS64 action to perform when the rule evaluates to TRUE. The built in actions function as follows:

* A default dns64 action with prefix <default prefix> and mapped and exclude are any

You can create custom actions by using the add dns action command in the CLI or the DNS64 > Actions > Create DNS64 Action dialog box in the NetScaler configuration utility.

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

boundTo

Location where policy is bound

labelType

Type of policy label invocation.

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

undefHits

Number of Undef hits.

description

Description of the policy

stateflag

devno

count

dns policylabel

The following operations can be performed on "dns policylabel":

[add](#) | [rm](#) | [bind](#) | [unbind](#) | [show](#) | [stat](#) | [rename](#)

add dns policylabel

Add a dns policy label.

Synopsis

```
add dns policylabel <labelName> <transform>
```

Arguments

labelName

Name of the dns policy label.

transform

The type of transformations allowed by the policies bound to the label.

Possible values: dns_req, dns_res

Example

```
add dns policylabel trans_dns dns_req
```

rm dns policylabel

Remove a dns policy label.

Synopsis

```
rm dns policylabel <labelName>
```

Arguments

labelName

Name of the dns policy label.

Example

```
rm dns policylabel trans_dns
```

bind dns policylabel

Bind the dns policy to one of the labels.

Synopsis

```
bind dns policylabel <labelName> <policyName> <priority> [<gotoPriorityExpression>] [-invoke (<labelType>  
<labelName>)]
```

Arguments

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

policyName

The dns policy name.

priority

Priority with which the policy is to be bound.

Minimum value: 1

Maximum value: 2147483647

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

o If gotoPriorityExpression is not present or if it is equal to END then the policy bank evaluation ends here

o Else if the gotoPriorityExpression is equal to NEXT then the next policy in the priority order is evaluated.

o Else gotoPriorityExpression is evaluated. The result of gotoPriorityExpression (which has to be a number) is processed as follows:

- An UNDEF event is triggered if

. gotoPriorityExpression cannot be evaluated

. gotoPriorityExpression evaluates to number which is smaller than the maximum priority in the policy bank but is not same as any policy's priority

. gotoPriorityExpression evaluates to a priority that is smaller than the current policy's priority

- If the gotoPriorityExpression evaluates to the priority of the current policy then the next policy in the priority order is evaluated.

- If the gotoPriorityExpression evaluates to the priority of a policy further ahead in the list then that policy will be evaluated next.

invoke

Invoke flag.

labelType

Type of policy label invocation.

Possible values: policylabel

Example

```
i) bind dns policylabel trans_dns pol_1 1 2 -invoke reqvserver CURRENT ii) bind rewrite
```

unbind dns policylabel

Unbind entities from dns label.

Synopsis

```
unbind dns policylabel <labelName> <policyName> [-priority <positive_integer>]
```

Arguments

labelName

Name of the dns policy label.

policyName

The dns policy name.

priority

Priority of the NOPOLICY to be unbound.

Minimum value: 1

Maximum value: 2147483647

Example

```
unbind dns policylabel trans_dns pol_1
```

show dns policylabel

Display policy label or policies bound to dns policylabel.

Synopsys

```
show dns policylabel [<labelName>]
```

Arguments

labelName

Name of the dns policy label.

Outputs

stateflag

transform

The type of transformations allowed by the policies bound to the label.

numpol

Number of polices bound to label.

hits

Number of times policy label was invoked.

policyName

The dns policy name.

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

invoke

Invoke flag.

labelType

Type of policy label invocation.

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

flowType

Flowtype of the bound dns policy.

description

Description of the policylabel

isDefault

A value of true is returned if it is a default dns policylabel.

flags

devno

count

Example

```
i) show dns policylabel trans_dns ii) show dns policylabel
```

stat dns policylabel

Display statistics of dns policylabel(s).

Synopsys

```
stat dns policylabel [<labelName>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

labelName

The name of the dns policy label for which statistics will be displayed. If not given statistics are shown for all dns policylabels.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Policy Label Hits (Hits)

Number of times policy label was invoked.

rename dns policylabel

Rename a dns policy label.

Synopsys

```
rename dns policylabel <labelName>@ <newName>@
```

Arguments

labelName

The name of the dns policylabel.

newName

The new name of the dns policylabel.

Example

```
rename dns policylabel oldname newname
```


dns proxyRecords

The following operations can be performed on "dns proxyRecords":

flush dns proxyRecords

Flushes all the proxy records from the DNS cache on the NetScaler appliance.

Synopsys

flush dns proxyRecords

dns ptrRec

The following operations can be performed on "dns ptrRec":

[add](#) | [rm](#) | [show](#)

add dns ptrRec

Creates a pointer (PTR) record for the specified reverse domain name.

Synopsis

```
add dns ptrRec <reverseDomain> <domain> ... [-TTL <secs>]
```

Arguments

reverseDomain

Reversed domain name representation of the IPv4 or IPv6 address for which to create the PTR record. Use the "in-addr.arpa." suffix for IPv4 addresses and the "ip6.arpa." suffix for IPv6 addresses.

domain

Domain name for which to configure reverse mapping.

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

Default value: 3600

Maximum value: 2147483647

Example

```
add dns ptrrec 1.1.1.in-addr.arpa. abc.com
```

rm dns ptrRec

Removes a pointer (PTR) record for the specified domain name and reverse domain name.

Synopsis

```
rm dns ptrRec <reverseDomain> [<domain> ...]
```

Arguments

reverseDomain

Reverse domain name of the PTR record.

domain

Domain name for which to remove reverse mapping.

Example

```
rm dns ptrrec 1.1.1.1.in-addr.arpa. ptr.com
```

show dns ptrRec

Displays the pointer (PTR) record for the specified reverse domain name and domain name.

Synopsys

```
show dns ptrRec [<reverseDomain> | -type <type>]
```

Arguments

reverseDomain

Reversed domain name representation of the IPv4 or IPv6 address for which to create the PTR record. Use the "in-addr.arpa." suffix for IPv4 addresses and the "ip6.arpa." suffix for IPv6 addresses.

type

Type of records to display. Available settings function as follows:

- * ADNS - Display all authoritative address records.
- * PROXY - Display all proxy address records.
- * ALL - Display all address records.

Possible values: ALL, ADNS, PROXY

Outputs

domain

Domain name for which to configure reverse mapping.

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

authType

Authentication type.

devno

count

stateflag

dns records

The following operations can be performed on "dns records":

stat dns records

Displays statistics for the specified DNS record or query type. If a DNS record or query type is not specified, statistics for all record and query types are shown.

Synopsys

```
stat dns records [<dnsRecordType>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

dnsRecordType

Display statistics for the specified DNS record or query type or, if a record or query type is not specified, statistics for all record types supported on the NetScaler appliance.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Total entries (totEntries)

Total number of DNS record entries

Total updates (totUpdates)

Total number of DNS proactive updates

Total responses (totResp)

Total number of DNS server responses

Total requests (totReq)

Total number of DNS queries recieved

Current entries (curEnt)

Current number of DNS entries

Total limit errors (errLim)

Total number of times we have recieved dns record with more entries than we support

Total response format errors (errRespFor)

Total number of times we have recieved malformed responses from the backend

Total alias exist errors (errAlias)

Total number of times we have recieved non-cname record for a domain for which an alias exists

Total cache misses (errNoDom)

Total number of cache misses

Current records (curRec)

Current number of DNS Records

dns soaRec

The following operations can be performed on "dns soaRec":

add | **rm** | **set** | **unset** | **show**

add dns soaRec

Creates a Start of Authority (SOA) record. Note: You can set the SOA parameters that are associated with zone transfers. However, the NetScaler appliance currently does not support zone transfers.

Synopsys

```
add dns soaRec <domain> -originServer <string> -contact <string> [-serial <positive_integer>] [-refresh <secs>] [-retry <secs>] [-expire <secs>] [-minimum <secs>] [-TTL <secs>]
```

Arguments

domain

Domain name for which to add the SOA record.

originServer

Domain name of the name server that responds authoritatively for the domain.

contact

Email address of the contact to whom domain issues can be addressed. In the email address, replace the @ sign with a period (.). For example, enter domainadmin.example.com instead of domainadmin@example.com.

serial

The secondary server uses this parameter to determine whether it requires a zone transfer from the primary server.

Default value: 100

Minimum value: 0

Maximum value: 4294967294

refresh

Time, in seconds, for which a secondary server must wait between successive checks on the value of the serial number.

Default value: 3600

Maximum value: 4294967294

retry

Time, in seconds, between retries if a secondary server's attempt to contact the primary server for a zone refresh fails.

Default value: 3

Maximum value: 4294967294

expire

Time, in seconds, after which the zone data on a secondary name server can no longer be considered authoritative because all refresh and retry attempts made during the period have failed. After the expiry period, the secondary server stops serving the zone. Typically one week. Not used by the primary server.

Default value: 3600

Maximum value: 4294967294

minimum

Default time to live (TTL) for all records in the zone. Can be overridden for individual records.

Default value: 5

Maximum value: 2147483647

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

Default value: 3600

Maximum value: 2147483647

rm dns soaRec

Removes the Start of Authority (SOA) record for the specified domain name.

Synopsys

```
rm dns soaRec <domain>
```

Arguments

domain

Domain name of the SOA record.

set dns soaRec

Modifies the parameters of the specified Start Of Authority (SOA) record.

Synopsys

```
set dns soaRec <domain> [-originServer <string>] [-contact <string>] [-serial <positive_integer>] [-refresh <secs>] [-retry <secs>] [-expire <secs>] [-minimum <secs>] [-TTL <secs>]
```

Arguments

domain

Domain of the SOA record to be modified.

originServer

Domain name of the name server that responds authoritatively for the domain.

contact

Email address of the contact to whom domain issues can be addressed. In the email address, replace the @ sign with a period (.). For example, enter domainadmin.example.com instead of domainadmin@example.com.

serial

The secondary server uses this parameter to determine whether it requires a zone transfer from the primary server.

Default value: 100

Minimum value: 1

Maximum value: 4294967294

refresh

Time, in seconds, for which a secondary server must wait between successive checks on the value of the serial number.

Default value: 3600

Maximum value: 4294967294

retry

Time, in seconds, between retries if a secondary server's attempt to contact the primary server for a zone refresh fails.

Default value: 3

Maximum value: 4294967294

expire

Time, in seconds, after which the zone data on a secondary name server can no longer be considered authoritative because all refresh and retry attempts made during the period have failed. After the expiry period, the secondary server stops serving the zone. Typically one week. Not used by the primary server.

Default value: 3600

Maximum value: 4294967294

minimum

Default time to live (TTL) for all records in the zone. Can be overridden for individual records.

Default value: 5

Maximum value: 2147483647

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

Default value: 3600

Maximum value: 2147483647

unset dns soaRec

Use this command to remove dns soaRec settings. Refer to the set dns soaRec command for meanings of the arguments.

Synopsys

```
unset dns soaRec <domain> [-serial] [-refresh] [-retry] [-expire] [-minimum] [-TTL]
```

show dns soaRec

Displays the parameters of the specified Start of Authority (SOA) record. If no domain name is specified, all SOA records are displayed.

Synopsys

show dns soaRec [<domain> | -type <type>]

Arguments

domain

The domain name.

type

Type of records to display. Available settings function as follows:

* ADNS - Display all authoritative address records.

* PROXY - Display all proxy address records.

* ALL - Display all address records.

Possible values: ALL, ADNS, PROXY

Outputs

originServer

Domain name of the name server that responds authoritatively for the domain.

contact

Email address of the contact to whom domain issues can be addressed. In the email address, replace the @ sign with a period (.). For example, enter domainadmin.example.com instead of domainadmin@example.com.

serial

The secondary server uses this parameter to determine whether it requires a zone transfer from the primary server.

refresh

Time, in seconds, for which a secondary server must wait between successive checks on the value of the serial number.

retry

Time, in seconds, between retries if a secondary server's attempt to contact the primary server for a zone refresh fails.

expire

Time, in seconds, after which the zone data on a secondary name server can no longer be considered authoritative because all refresh and retry attempts made during the period have failed. After the expiry period, the secondary server stops serving the zone. Typically one week. Not used by the primary server.

minimum

Default time to live (TTL) for all records in the zone. Can be overridden for individual records.

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

authType

Record type.

devno

count

stateflag

dns srvRec

The following operations can be performed on "dns srvRec":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add dns srvRec

Creates a service (SRV) record for the service offered by the specified target host, in the specified domain.

Synopsys

```
add dns srvRec <domain> <target> -priority <positive_integer> -weight <positive_integer> -port <positive_integer> [-TTL <secs>]
```

Arguments

domain

Domain name, which, by convention, is prefixed by the symbolic name of the desired service and the symbolic name of the desired protocol, each with an underscore (_) prepended. For example, if an SRV-aware client wants to discover a SIP service that is provided over UDP, in the domain example.com, the client performs a lookup for _sip._udp.example.com.

target

Target host for the specified service.

priority

Integer specifying the priority of the target host. The lower the number, the higher the priority. If multiple target hosts have the same priority, selection is based on the Weight parameter.

Minimum value: 0

Maximum value: 65535

weight

Weight for the target host. Aids host selection when two or more hosts have the same priority. A larger number indicates greater weight.

Minimum value: 0

Maximum value: 65535

port

Port on which the target host listens for client requests.

Minimum value: 0

Maximum value: 65535

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

Default value: 3600

Maximum value: 2147483647

rm dns srvRec

Removes, from the specified domain, the SRV record created for the service provided by the specified target host.

Synopsys

```
rm dns srvRec <domain> <target> ...
```

Arguments

domain

Domain name of the SRV record.

target

Target host for the specified service.

set dns srvRec

Modifies the parameters of the specified service (SRV) record.

Synopsys

```
set dns srvRec <domain> <target> [-priority <positive_integer>] [-weight <positive_integer>] [-port  
<positive_integer>] [-TTL <secs>]
```

Arguments

domain

Name of the SRV record to be modified.

target

Target of the SRV record to be modified.

priority

Integer specifying the priority of the target host. The lower the number, the higher the priority. If multiple target hosts have the same priority, selection is based on the Weight parameter.

Minimum value: 0

Maximum value: 65535

weight

Weight for the target host. Aids host selection when two or more hosts have the same priority. A larger number indicates greater weight.

Minimum value: 0

Maximum value: 65535

port

Port on which the target host listens for client requests.

Minimum value: 0

Maximum value: 65535

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the

domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

Default value: 3600

Maximum value: 2147483647

unset dns srvRec

Use this command to remove dns srvRec settings. Refer to the set dns srvRec command for meanings of the arguments.

Synopsys

```
unset dns srvRec <domain> <target> -TTL
```

show dns srvRec

Displays the service (SRV) record configured for the specified target host and domain. If the domain name is not specified, all of the SRV records are shown.

Synopsys

```
show dns srvRec [(<domain> [<target>]) | -type <type>]
```

Arguments

domain

Domain name for which to display the SRV record.

target

Target host for the specified service.

type

Type of records to display. Available settings function as follows:

* ADNS - Display all authoritative address records.

* PROXY - Display all proxy address records.

* ALL - Display all address records.

Possible values: ALL, ADNS, PROXY

Outputs

priority

Priority of the target host. This helps in server selection by the client.

weight

Weight for the target host. Aids host selection when two or more hosts have the same priority. A larger number indicates greater weight.

port

Port on which the target host listens for client requests.

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong

to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

authType

Record type.

devno

count

stateflag

dns stats

The following operations can be performed on "dns stats":

show dns stats

show dns stats is an alias for stat dns

Synopsys

show dns stats - alias for 'stat dns'

dns suffix

The following operations can be performed on "dns suffix":

[add](#) | [rm](#) | [show](#)

add dns suffix

Specifies a suffix that can be used to complete domain names that are not fully qualified. For example, if you specify the example.com suffix, and the NetScaler appliance is required to resolve the incomplete domain name "myhost," it attempts to resolve "myhost.example.com."

Synopsys

```
add dns suffix <dnsSuffix>
```

Arguments

dnsSuffix

Suffix to be appended when resolving domain names that are not fully qualified.

Example

```
add dns suffix netscaler.com  If the incoming domain name "engineering" is not resolved by
```

rm dns suffix

Removes a DNS suffix.

Synopsys

```
rm dns suffix <dnsSuffix>
```

Arguments

dnsSuffix

DNS suffix to remove.

show dns suffix

Displays the specified DNS suffix or, if no DNS suffix is specified, all configured DNS suffixes.

Synopsys

```
show dns suffix [<dnsSuffix>]
```

Arguments

dnsSuffix

DNS suffix to display.

Outputs

devno

count

stateflag

dns txtRec

The following operations can be performed on "dns txtRec":

[add](#) | [rm](#) | [show](#)

add dns txtRec

Creates a text (TXT) record for the specified domain name. Each resource record is stored with a unique, internally generated record ID, which you can view and use to delete the record. You cannot modify a TXT resource record.

Synopsys

```
add dns txtRec <domain> <string> ... [-TTL <secs>]
```

Arguments

domain

Name of the domain for the TXT record.

string

Information to store in the TXT resource record. Enclose the string in single or double quotation marks. A TXT resource record can contain up to six strings, each of which can contain up to 255 characters. If you want to add a string of more than 255 characters, evaluate whether splitting it into two or more smaller strings, subject to the six-string limit, works for you.

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

Default value: 3600

Maximum value: 2147483647

Example

```
add dns txtRec spf.m.test. "v=spf1 ip4:1.2.3.0/24 ip4:1.3.4.0/24 ?all" add dns txtRec cor
```

rm dns txtRec

Removes the specified TXT record from the specified domain.

Synopsys

```
rm dns txtRec <domain> (<string> ... | -recordId <positive_integer>@)
```

Arguments

domain

Name of the domain for the TXT record.

string

Complete set of text strings in the TXT record, entered in the order in which they are stored in the record. Mutually exclusive with the record ID parameter.

recordId

Unique, internally generated record ID. View the details of the TXT record to obtain its record ID. Mutually exclusive with the string parameter.

Minimum value: 1

Maximum value: 65535

Example

```
rm dns txtRec spf.m.test. "v=spf1 ip4:1.2.3.0/24 ip4:1.3.4.0/24 ?all" rm dns txtRec comm
```

show dns txtRec

Displays TXT records owned by the specified domain. If no domain name is specified, all configured TXT records are shown.

Synopsys

```
show dns txtRec [<domain> | -type <type>]
```

Arguments

domain

Name of the domain for the TXT record.

type

Type of records to display. Available settings function as follows:

* ADNS - Display all authoritative address records.

* PROXY - Display all proxy address records.

* ALL - Display all address records.

Possible values: ALL, ADNS, PROXY

Default value: ADNS

Outputs

string

Information to store in the TXT resource record. Enclose the string in single or double quotation marks. A TXT resource record can contain up to six strings, each of which can contain up to 255 characters. If you want to add a string of more than 255 characters, evaluate whether splitting it into two or more smaller strings, subject to the six-string limit, works for you.

TTL

Time to Live (TTL), in seconds, for the record. TTL is the time for which the record must be cached by DNS proxies. The specified TTL is applied to all the resource records that are of the same record type and belong to the specified domain name. For example, if you add an address record, with a TTL of 36000, to the domain name example.com, the TTLs of all the address records of example.com are changed to 36000. If the TTL is not specified, the NetScaler appliance uses either the DNS zone's minimum TTL or, if the SOA record is not available on the appliance, the default value of 3600.

recordId

authType

Authentication type.

devno

count

stateflag

Example

```
show dns txtRec spf.m.test. show dns txtRec
```

dns view

The following operations can be performed on "dns view":

[add](#) | [rm](#) | [show](#)

add dns view

Creates a DNS view. A DNS view is used in global server load balancing (GSLB) to return a predetermined IP address to a specific group of clients, which are identified by using a DNS policy.

Synopsys

```
add dns view <viewName>
```

Arguments

viewName

Name for the DNS view.

Example

```
add dns view privateview
```

rm dns view

Removes a DNS view.

Synopsys

```
rm dns view <viewName>
```

Arguments

viewName

Name for the DNS view.

Example

```
rm dns view privateview
```

show dns view

Displays the specified DNS view or, if no DNS view name is specified, all the DNS views configured on the NetScaler appliance.

Synopsys

```
show dns view [<viewName>]
```

Arguments

viewName

Name of the view to display.

Outputs

serviceName

Service name of the service using this view.

gslbServiceName

Service name of the service using this view.

dnsPolicyName

dnspolicy name of this view.

IPAddress

IP of the service corresponding to the given view.

flags

Flags controlling display.

stateflag

flags controlling display

devno

count

dns zone

The following operations can be performed on "dns zone":

add | **set** | **unset** | **rm** | **sign** | **unsign** | **show**

add dns zone

Creates a DNS zone on the NetScaler appliance. Mandatory if you want to use the appliance to implement Domain Name Security Extensions (DNSSEC) for the zone. When you add a DNS resource record, if the domain name of the record belongs to the zone, the record is automatically added to the zone.

Synopsys

```
add dns zone <zoneName> -proxyMode ( YES | NO ) [-dnssecOffload ( ENABLED | DISABLED ) [-nsec ( ENABLED | DISABLED )]]
```

Arguments

zoneName

Name of the zone to create.

proxyMode

Deploy the zone in proxy mode. Enable in the following scenarios:

- * The load balanced DNS servers are authoritative for the zone and all resource records that are part of the zone.

- * The load balanced DNS servers are authoritative for the zone, but the NetScaler appliance owns a subset of the resource records that belong to the zone (partial zone ownership configuration). Typically seen in global server load balancing (GSLB) configurations, in which the appliance responds authoritatively to queries for GSLB domain names but forwards queries for other domain names in the zone to the load balanced servers.

In either scenario, do not create the zone's Start of Authority (SOA) and name server (NS) resource records on the appliance.

Disable if the appliance is authoritative for the zone, but make sure that you have created the SOA and NS records on the appliance before you create the zone.

Possible values: YES, NO

Default value: ENABLED

dnssecOffload

Enable dnssec offload for this zone.

Possible values: ENABLED, DISABLED

Default value: DISABLED

nsec

Enable nsec generation for dnssec offload.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
add dns zone foo.bar -proxyMode NO -dnssec ENABLED
```

set dns zone

Modifies the parameters of the specified DNS zone.

Synopsys

```
set dns zone <zoneName> [-proxyMode ( YES | NO )] [-dnssecOffload ( ENABLED | DISABLED )] [-nsec ( ENABLED | DISABLED )]
```

Arguments

zoneName

Name of the zone.

proxyMode

Deploy the zone in proxy mode. Enable in the following scenarios:

- * The load balanced DNS servers are authoritative for the zone and all resource records that are part of the zone.

- * The load balanced DNS servers are authoritative for the zone, but the NetScaler appliance owns a subset of the resource records that belong to the zone (partial zone ownership configuration). Typically seen in global server load balancing (GSLB) configurations, in which the appliance responds authoritatively to queries for GSLB domain names but forwards queries for other domain names in the zone to the load balanced servers.

In either scenario, do not create the zone's Start of Authority (SOA) and name server (NS) resource records on the appliance.

Disable if the appliance is authoritative for the zone, but make sure that you have created the SOA and NS records on the appliance before you create the zone.

Possible values: YES, NO

Default value: ENABLED

dnssecOffload

Enable dnssec offload for this zone.

Possible values: ENABLED, DISABLED

Default value: DISABLED

nsec

Enable nsec generation for dnssec offload.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
set dns zone foo.bar -proxyMode NO -dnssec ENABLED
```

unset dns zone

Use this command to remove dns zone settings. Refer to the set dns zone command for meanings of the arguments.

Synopsys

```
unset dns zone <zoneName> [-proxyMode] [-dnssecOffload] [-nsec]
```

rm dns zone

Removes a DNS zone from the NetScaler appliance.

Synopsys

```
rm dns zone <zoneName>
```

Arguments

zoneName

Name of the zone to remove.

sign dns zone

Signs a DNS zone with a DNS key. Before you sign a zone, make sure that you've enabled DNSSEC by setting the global DNS parameter "Enable DNSSEC extension."

Synopsys

```
sign dns zone <zoneName> [-keyName <string> ...]
```

Arguments

zoneName

Name of the zone.

keyName

Name of the public/private DNS key pair with which to sign the zone. You can sign a zone with up to four keys.

Example

```
sign dns zone abc.com. -keyname abc.com.zsk abc.com.ksk
```

unsign dns zone

Unsigns the specified DNS zone with the specified DNS key.

Synopsys

```
unsign dns zone <zoneName> [-keyName <string> ...]
```

Arguments

zoneName

Name of the zone.

keyName

Name of the public-private DNS key pair with which to unsign the zone.

Example

```
unsign dns zone abc.com. -keyname abc.com.zsk abc.com.ksk
```

show dns zone

Displays the parameters of the specified DNS zone, along with information about the types of resource records available for each domain name in the zone. If no zone name is specified, just the parameters are shown, for all configured zones.

Synopsys

show dns zone [<zoneName> | -type <type>]

Arguments

zoneName

Name of the zone. Mutually exclusive with the type parameter.

type

Type of zone to display. Mutually exclusive with the DNS Zone (zoneName) parameter. Available settings function as follows:

- * ADNS - Display all the zones for which the NetScaler appliance is authoritative.
- * PROXY - Display all the zones for which the NetScaler appliance is functioning as a proxy server.
- * ALL - Display all the zones configured on the appliance.

Possible values: ALL, ADNS, PROXY

Outputs

proxyMode

Deploy the zone in proxy mode. Enable in the following scenarios:

* The load balanced DNS servers are authoritative for the zone and all resource records that are part of the zone.

* The load balanced DNS servers are authoritative for the zone, but the NetScaler appliance owns a subset of the resource records that belong to the zone (partial zone ownership configuration). Typically seen in global server load balancing (GSLB) configurations, in which the appliance responds authoritatively to queries for GSLB domain names but forwards queries for other domain names in the zone to the load balanced servers.

In either scenario, do not create the zone's Start of Authority (SOA) and name server (NS) resource records on the appliance.

Disable if the appliance is authoritative for the zone, but make sure that you have created the SOA and NS records on the appliance before you create the zone.

flags

Flags controlling display.

nsecBitarray

Bit array representing the different record types configured for the domain name

domain

Domain name that belongs to the given zone

nextRecs

An array of record types associated with the nsec record.

stateflag

flags controlling display

dnssecOffload

Enable dnssec offload for this zone.

nsec

Enable nsec generation for dnssec offload.

keyName

Name of the public/private DNS key pair with which to sign the zone. You can sign a zone with up to four keys.

sigInceptionTime

The time when sign was done with this key.

signed

Integer which denote status of keys.

expires

Time period for which to consider the key valid, after the key is used to sign a zone.

devno**count**

Example

```
show dns zone foo.bar
```

DOS Commands

The entities on which you can perform NetScaler CLI operations:

- dos
- dos policy
- dos stats

dos

The following operations can be performed on "dos":

stat dos

Displays DoS protection statistics.

Synopsys

```
stat dos [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

DOS condition triggered (CndMatch)

Number of times the NetScaler appliance triggered the DOS JavaScript due to a condition match.

Valid DOS clients (ValidClt)

Number of clients from whom the NetScaler appliance received a valid DOS cookie.

DOS priority clients (DosPriCl)

Number of valid clients that were given DOS priority.

dos policy

The following operations can be performed on "dos policy":

add | **rm** | **set** | **unset** | **show** | **stat**

add dos policy

Adds a DoS protection policy to the appliance. Note: To apply DoS protection to a service, bind the DoS policy to the service by using the bind service command.

Synopsys

```
add dos policy <name> -qDepth <positive_integer> [-cltDetectRate <positive_integer>]
```

Arguments

name

Name for the HTTP DoS protection policy. Must begin with a letter, number, or the underscore character (_). Other characters allowed, after the first character, are the hyphen (-), period (.) hash (#), space (), at (@), equals (=), and colon (:) characters.

qDepth

Queue depth. The queue size (the number of outstanding service requests on the system) before DoS protection is activated on the service to which the DoS protection policy is bound.

Minimum value: 21

cltDetectRate

Client detect rate. Integer representing the percentage of traffic to which the HTTP DoS policy is to be applied after the queue depth condition is satisfied.

Minimum value: 0

Maximum value: 100

Example

```
add dos policy dospol -qdepth 100 -cltDetectRate 90
```

rm dos policy

Removes a DoS protection policy from the appliance.

Synopsys

```
rm dos policy <name>
```

Arguments

name

Name of the DoS protection policy to be removed.

Example

```
rm dos policy dospol
```

set dos policy

Modifies the attributes of a DoS protection policy.

Synopsys

```
set dos policy <name> [-qDepth <positive_integer>] [-cltDetectRate <positive_integer>]
```

Arguments

name

Name of the DoS protection policy to be modified.

qDepth

Queue depth. The queue size (the number of outstanding service requests on the system) before DoS protection is activated on the service to which the DoS protection policy is bound.

Minimum value: 21

cltDetectRate

Client detect rate. Integer representing the percentage of traffic to which the HTTP DoS policy is to be applied after the queue depth condition is satisfied.

Minimum value: 1

Maximum value: 100

Example

```
set dos policy dospol -qdepth 1000
```

unset dos policy

Use this command to remove dos policy settings. Refer to the set dos policy command for meanings of the arguments.

Synopsys

```
unset dos policy <name> -cltDetectRate
```

show dos policy

Displays information about a DoS protection policy.

Synopsys

```
show dos policy [<name>]
```

Arguments

name

Name of the DoS protection policy about which to display information. If a name is not provided, information about all DoS protection policies is shown.

Outputs

qDepth

Queue depth. The queue size (the number of outstanding service requests on the system) before DoS protection is activated on the service to which the DoS protection policy is bound.

cltDetectRate

Client detect rate. Integer representing the percentage of traffic to which the HTTP DoS policy is to be applied after the queue depth condition is satisfied.

devno

count

stateflag

Example

```
> show dos policy          1 configured DoS policy: 1)          Policy: dospol  QDepth: 100
```

stat dos policy

Displays statistics of the DoS protection policy.

Synopsys

stat dos policy [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

name

The name of the DoS protection policy whose statistics must be displayed. If a name is not provided, statistics of all the DoS protection policies are displayed.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Client detect rate (CIDtRate)

Current ratio of JavaScript send rate to the server response rate (Client detect rate)

Physical service IP (SvcIP)

IP address of the service to which this policy is bound.

Physical service port (SvcPort)

Port address of the service to which this policy is bound.

Current server queue size (CurQSize)

Current queue size of the server to which this policy is bound.

DOS transactions (DosTrans)

Total number of DoS JavaScript transactions performed for this policy.

Client detect rate mismatch (JsRefusd)

Number of times the DoS JavaScript was not sent because the set JavaScript rate was not met for this policy.

Valid clients (TotValCI)

Total number of valid DoS cookies received for this policy.

DOS JavaScript bytes served (JsBytSnt)

Total number of DoS JavaScript bytes sent for this policy.

Non GET, POST requests

Number of non-GET and non-POST requests for which DOS JavaScript was sent.

DOS JavaScript send rate (JSRate)

Current rate at which JavaScript is being sent in response to client requests.

Server response rate (RespRate)

Current rate at which the server to which this policy is bound is responding.

dos stats

The following operations can be performed on "dos stats":

show dos stats

show dos stats is an alias for stat dos Displays DoS protection statistics.

Synopsys

show dos stats - alias for 'stat dos'

Front End Optimization Commands

The entities on which you can perform NetScaler CLI operations:

- o feo
- o feo action
- o feo global
- o feo parameter
- o feo policy
- o feo stats

feo

The following operations can be performed on "feo":

stat feo

Shows front end optimization performance statistics.

Synopsys

```
stat feo [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

Optimized cache objects (VcDn)

Total number of optimized cache objects ready to be served.

Original cache objects (OcDn)

Total number of original cache objects ready to be served.

Domain Sharded (DmShd)

Total no of images whose domain has been set from shards.

Images lazy loaded (ImLzLd)

Total number of images modified for lazy loading.

URI Replaced (UrRep)

Total number of URI replaced.

Inlined Imgs in CSS (InCslm)

Total number of images inlined in CSS.

Inlined JS (InJs)

Total number of inlined JS files.

Inlined CSS (InCs)

Total number of inlined CSS files.

Inlined images (InIm)

Total number of inlined images in HTML.

Data Savings in KB (DatSav)

Total data savings in bytes.

HTML comments removed (HtmCmtR)

The total number of HTML comments removed.

Cache extended (CacExt)

The total number of objects cache extended.

CSS combined (CssComb)

The total number of CSS combined.

Import to Links (ImpToLink)

The total number of CSS imports converted to links

JS moved to end (JsMoved)

The total number of JS moved to end.

CSS moved to head (CssMoved)

The total number of CSS moved to head.

JS Minified (JsMin)

The total number of JS files minified.

CSS Minified (CssMin)

The total number of CSS files minified.

JPEGs Optimized (JpegsOpt)

The total number of JPEG format images optimized.

Gif to PNGs (GifToPng)

The total number of images converted from GIF to PNG format.

Images Resized (ImgResz)

The total number of images resized to dimensions in the tag.

feo action

The following operations can be performed on "feo action":

add | **set** | **unset** | **rm** | **show**

add feo action

Create a front end optimization action.

Synopsys

```
add feo action <name> [-pageExtendCache] [-imgShrinkToAttrib] [-imgGifToPng] [-imgInline] [-cssImgInline] [-jpgOptimize] [-imgLazyLoad] [-cssMinify] [-cssInline] [-cssCombine] [-convertImportToLink] [-jsMinify] [-jsInline] [-htmlMinify] [-cssMoveToHead] [-jsMoveToEND] [-domainSharding <string> <dnsShards> ...] [-clientSideMeasurements]
```

Arguments

name

The name of the front end optimization action.

pageExtendCache

Extend the time period during which the browser can use the cached resource.

imgShrinkToAttrib

Shrink image dimensions as per the height and width attributes specified in the tag.

imgGifToPng

Convert GIF image formats to PNG formats.

imgInline

Inline images whose size is less than 2KB.

cssImgInline

Inline small images (less than 2KB) referred within CSS files as background-URLs

jpgOptimize

Remove non-image data such as comments from JPEG images.

imgLazyLoad

Download images, only when the user scrolls the page to view them.

cssMinify

Remove comments and whitespaces from CSSs.

cssInline

Inline CSS files, whose size is less than 2KB, within the main page.

cssCombine

Combine one or more CSS files into one file.

convertImportToLink

Convert CSS import statements to HTML link tags.

jsMinify

Remove comments and whitespaces from JavaScript.

jsInline

Convert linked JavaScript files (less than 2KB) to inline JavaScript files.

htmlMinify

Remove comments and whitespaces from an HTML page.

cssMoveToHead

Move any CSS file present within the body tag of an HTML page to the head tag.

jsMoveToEnd

Move any JavaScript present in the body tag to the end of the body tag.

domainSharding

Domain name of the server

dnsShards

Set of domain names that replaces the parent domain.

clientSideMeasurements

Collect the amount of time required for the client to load and render the web page.

set feo action

Modify a front end optimization action.

Synopsys

```
set feo action <name> [-pageExtendCache] [-imgShrinkToAttrib] [-imgGifToPng] [-imgInline] [-cssImgInline] [-jpgOptimize] [-imgLazyLoad] [-cssMinify] [-cssInline] [-cssCombine] [-convertImportToLink] [-jsMinify] [-jsInline] [-htmlMinify] [-cssMoveToHead] [-jsMoveToEnd] [-domainSharding <string> <dnsShards> ...] [-clientSideMeasurements]
```

Arguments

name

The name of the front end optimization action.

pageExtendCache

Extend the time period during which the browser can use the cached resource.

imgShrinkToAttrib

Shrink image dimensions as per the height and width attributes specified in the tag.

imgGifToPng

Convert GIF image formats to PNG formats.

imgInline

Inline images whose size is less than 2KB.

cssImgInline

Inline small images (less than 2KB) referred within CSS files as background-URLs

jpgOptimize

Remove non-image data such as comments from JPEG images.

imgLazyLoad

Download images, only when the user scrolls the page to view them.

cssMinify

Remove comments and whitespaces from CSSs.

cssInline

Inline CSS files, whose size is less than 2KB, within the main page.

cssCombine

Combine one or more CSS files into one file.

convertImportToLink

Convert CSS import statements to HTML link tags.

jsMinify

Remove comments and whitespaces from JavaScript.

jsInline

Convert linked JavaScript files (less than 2KB) to inline JavaScript files.

htmlMinify

Remove comments and whitespaces from an HTML page.

cssMoveToHead

Move any CSS file present within the body tag of an HTML page to the head tag.

jsMoveToEND

Move any JavaScript present in the body tag to the end of the body tag.

domainSharding

Domain name of the server

dnsShards

Set of domain names that replaces the parent domain.

clientSideMeasurements

Collect the amount of time required for the client to load and render the web page.

unset feo action

Modify a front end optimization action..Refer to the set feo action command for meanings of the arguments.

Synopsys

```
unset feo action <name> [-pageExtendCache] [-imgShrinkToAttrib] [-imgGifToPng] [-imgInline] [-cssImgInline] [-jpgOptimize] [-imgLazyLoad] [-cssMinify] [-cssInline] [-cssCombine] [-convertImportToLink] [-jsMinify] [-jsInline] [-htmlMinify] [-cssMoveToHead] [-jsMoveToEND] [-clientSideMeasurements] [-domainSharding]
```

rm feo action

Remove the specified front end optimization action.

Synopsys

rm feo action <name>

Arguments

name

The name of the front end optimization action.

show feo action

Display the front end optimization actions defined, including the built-in actions.

Synopsys

show feo action [<name>]

Arguments

name

The name of the front end optimization action.

Outputs

stateflag

pageExtendCache

Extend the time period during which the browser can use the cached resource.

imgShrinkToAttrib

Shrink image dimensions as per the height and width attributes specified in the tag.

imgGifToPng

Convert GIF image formats to PNG formats.

imgAddDimensions

Add dimension attributes to images, if not specified within the tag.

imgShrinkForMobile

Serve smaller images for mobile users.

imgWeaken

Reduce the image quality.

imgInline

Inline images whose size is less than 2KB.

cssImgInline

Inline small images (less than 2KB) referred within CSS files as background-URLs

jpgOptimize

Remove non-image data such as comments from JPEG images.

jpgProgressive

Convert JPEG image formats to progressive formats.

imgLazyLoad

Download images, only when the user scrolls the page to view them.

cssMinify

Remove comments and whitespaces from CSSs.

cssInline

Inline CSS files, whose size is less than 2KB, within the main page.

cssCombine

Combine one or more CSS files into one file.

cssFlattenImports

Replace CSS import statements with the file content.

convertImportToLink

Convert CSS import statements to HTML link tags.

jsMinify

Remove comments and whitespaces from JavaScript.

jsInline

Convert linked JavaScript files (less than 2KB) to inline JavaScript files.

jsCombine

Combine one or more JavaScript files into one file.

htmlMinify

Remove comments and whitespaces from an HTML page.

htmlRmDefaultAttribs

Remove default redundant attributes from an HTML file.

htmlRmAttribQuotes

Remove unnecessary quotes present within the HTML attributes.

htmlTrimUrls

Trim URLs.

cssMoveToHead

Move any CSS file present within the body tag of an HTML page to the head tag.

jsMoveToEND

Move any JavaScript present in the body tag to the end of the body tag.

domainSharding

Domain name of the server

dnsShards

Set of domain names that replaces the parent domain.

clientSideMeasurements

Collect the amount of time required for the client to load and render the web page.

hits

The number of times the action has been taken.

undefHits

Total number of undefined policy hits.

builtin

Flag to determine if front end optimization action is built-in or not.

devno**count**

feo global

The following operations can be performed on "feo global":

[bind](#) | [unbind](#) | [show](#)

bind feo global

Bind a front end optimization policy globally.

Synopsis

```
bind feo global <policyName> <priority> [-type <type>] [<gotoPriorityExpression>]
```

Arguments

policyName

Name of the front end optimization policy.

priority

The priority assigned to the policy binding.

Minimum value: 1

Maximum value: 2147483647

type

Bind point, specifying where to bind the policy. This is relevant for advanced (default-syntax) policies only.

Possible values: REQ_OVERRIDE, REQ_DEFAULT, RES_OVERRIDE, RES_DEFAULT, NONE

Default value: NS_REQ_DEFAULT

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

o If gotoPriorityExpression is not present or if it is equal to END then the policy bank evaluation ends here

o Else if the gotoPriorityExpression is equal to NEXT then the next policy in the priority order is evaluated.

o Else gotoPriorityExpression is evaluated. The result of gotoPriorityExpression (which has to be a number) is processed as follows:

- An UNDEF event is triggered if

- . gotoPriorityExpression cannot be evaluated

- . gotoPriorityExpression evaluates to number which is smaller than the maximum priority in the policy bank but is not same as any policy's priority

- . gotoPriorityExpression evaluates to a priority that is smaller than the current policy's priority

- If the gotoPriorityExpression evaluates to the priority of the current policy then the next policy in the priority order is evaluated.

- If the gotoPriorityExpression evaluates to the priority of a policy further ahead in the list then that policy will be evaluated next.

unbind feo global

Unbind a front end optimization policy globally.

Synopsys

unbind feo global <policyName> [-type <type> [-priority <positive_integer>]]

Arguments

policyName

Name of the front end optimization policy.

type

Bindpoint, specifying from where to unbind the policy. Applicable only to advanced (default-syntax) policies.

Possible values: REQ_OVERRIDE, REQ_DEFAULT, RES_OVERRIDE, RES_DEFAULT, NONE

priority

Priority of the policy to be unbound.

Minimum value: 1

Maximum value: 2147483647

show feo global

Display the globally bound front end optimization policies.

Synopsys

show feo global [-type <type>]

Arguments

type

Bindpoint to which the policy is bound.

Possible values: REQ_OVERRIDE, REQ_DEFAULT, RES_OVERRIDE, RES_DEFAULT, NONE

Outputs

stateflag

policyName

The name of the globally bound front end optimization policy.

priority

The priority assigned to the policy binding.

numpol

The number of policies bound to the bindpoint.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

devno

count

feo parameter

The following operations can be performed on "feo parameter":

[set](#) | [unset](#) | [show](#)

set feo parameter

Configure front end optimization parameters.

Synopsys

```
set feo parameter [-cacheMaxage <positive_integer>] [-JpegQualityPercent <positive_integer>] [-cssInlineThresSize <positive_integer>] [-jsInlineThresSize <positive_integer>] [-imgInlineThresSize <positive_integer>]
```

Arguments

cacheMaxage

Maximum period (in days), for cache extension.

Default value: 30

Minimum value: 0

Maximum value: 360

JpegQualityPercent

The percentage value of a JPEG image quality to be reduced. Range: 0 - 100

Default value: 75

Minimum value: 0

Maximum value: 100

cssInlineThresSize

Threshold value of the file size (in bytes) for converting external CSS files to inline CSS files.

Default value: 1024

Minimum value: 1

Maximum value: 2048

jsInlineThresSize

Threshold value of the file size (in bytes), for converting external JavaScript files to inline JavaScript files.

Default value: 1024

Minimum value: 1

Maximum value: 2048

imgInlineThresSize

Maximum file size of an image (in bytes), for coverting linked images to inline images.

Default value: 1024

Minimum value: 1

Maximum value: 2048

Example

```
set feo param -CacheMaxAge 8 -JpegQualityPercent 80 -cssInlineThresSize 1024 -jsInlineThresSize 1024
```

unset feo parameter

Use this command to remove feo parameter settings. Refer to the set feo parameter command for meanings of the arguments.

Synopsys

```
unset feo parameter [-cacheMaxAge] [-JpegQualityPercent] [-cssInlineThresSize] [-jsInlineThresSize] [-imgInlineThresSize]
```

show feo parameter

Display front end optimization parameters

Synopsys

```
show feo parameter
```

Outputs

cacheMaxage

Maximum period (in days), for cache extension.

JpegQualityPercent

The percentage value of a JPEG image quality to be reduced. Range: 0 - 100

cssInlineThresSize

Threshold value of the file size (in bytes) for converting external CSS files to inline CSS files.

jsInlineThresSize

Threshold value of the file size (in bytes), for converting external JavaScript files to inline JavaScript files.

imgInlineThresSize

Maximum file size of an image (in bytes), for converting linked images to inline images.

Example

```
show feo param
```

feo policy

The following operations can be performed on "feo policy":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add feo policy

Create a front end optimization policy.

Synopsys

add feo policy <name> <rule> <action>

Arguments

name

The name of the front end optimization policy.

rule

The rule associated with the front end optimization policy.

action

The front end optimization action that has to be performed when the rule matches.

rm feo policy

Remove a front end optimization policy.

Synopsys

rm feo policy <name>

Arguments

name

The front end optimization policy to be removed.

set feo policy

Modify a front end optimization policy.

Synopsys

set feo policy <name> [-rule <expression>] [-action <string>]

Arguments

name

The front end optimization policy to be modified.

rule

The new rule to be associated with the front end optimization policy.

action

The optimization to be associated with the front end optimization policy.

unset feo policy

Use this command to remove feo policy settings. Refer to the set feo policy command for meanings of the arguments.

Synopsys

```
unset feo policy <name> [-rule] [-action]
```

show feo policy

Display the configured front end optimization policies.

Synopsys

```
show feo policy [<name>]
```

Arguments

name

The name of the front end optimization policy.

Outputs

stateflag

rule

The rule associated with the front end optimization policy.

action

The front end optimization action that has to be performed when the rule matches.

builtin

Flag to determine if the front end optimization policy is built-in or not

hits

Total number of hits.

undefHits

Total number of undefined policy hits.

activePolicy

Indicates whether a policy is bound or not.

boundTo

Location where the policy is bound to.

priority

Priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

devno

count

feo stats

The following operations can be performed on "feo stats":

show feo stats

show feo stats is an alias for stat feo Displays Front end optimization statistics.

Synopsys

show feo stats - alias for 'stat feo'

Filter Commands

The entities on which you can perform NetScaler CLI operations:

- o filter action
- o filter global
- o filter htmlinjectionparameter
- o filter htmlinjectionvariable
- o filter policy
- o filter postbodyInjection
- o filter prebodyInjection

filter action

The following operations can be performed on "filter action":

add | **rm** | **set** | **unset** | **show**

add filter action

Creates a content filtering action. This action can be associated with a content filtering policy that is created with the add filter policy command. Note: The following content filtering actions are available by default: * RESET - Sends a TCP reset for the HTTP requests. * DROP - Drops the HTTP requests silently, without sending a TCP FIN for closing the connection.

Synopsys

add filter action <name> <qual> [<serviceName>] [<value>] [<respCode>] [<page>]

Arguments

name

Name for the filtering action. Must begin with a letter, number, or the underscore character (_). Other characters allowed, after the first character, are the hyphen (-), period (.), hash (#), space (), at sign (@), equals (=), and colon (:) characters. Choose a name that helps identify the type of action. The name of a filter action cannot be changed after it is created.

CLI Users: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my action" or 'my action').

qual

Qualifier, which is the action to be performed. The qualifier cannot be changed after it is set. The available options function as follows:

ADD - Adds the specified HTTP header.

RESET - Terminates the connection, sending the appropriate termination notice to the user's browser.

FORWARD - Redirects the request to the designated service. You must specify either a service name or a page, but not both.

DROP - Silently deletes the request, without sending a response to the user's browser.

CORRUPT - Modifies the designated HTTP header to prevent it from performing the function it was intended to perform, then sends the request/response to the server/browser.

ERRORCODE. Returns the designated HTTP error code to the user's browser (for example, 404, the standard HTTP code for a non-existent Web page).

Possible values: reset, add, corrupt, forward, errorcode, drop

serviceName

Service to which to forward HTTP requests. Required if the qualifier is FORWARD.

value

String containing the header_name and header_value. If the qualifier is ADD, specify <header_name>: <header_value>. If the qualifier is CORRUPT, specify only the header_name

respCode

Response code to be returned for HTTP requests (for use with the ERRORCODE qualifier).

Minimum value: 1

page

HTML page to return for HTTP requests (For use with the ERRORCODE qualifier).

Example

```
add filter action bad_url_action errorcode 400 "<HTML>Bad URL.</HTML>" add filter action :
```

rm filter action

Removes a content filtering action.

Synopsis

```
rm filter action <name>
```

Arguments

name

Name of the content filter action to be removed.

Example

```
rm filter action filter_action_name
```

set filter action

Modifies an existing content filtering action.

Synopsis

```
set filter action <name> [-serviceName <string>] [-value <string>] [-respCode <positive_integer>] [-page <string>]
```

Arguments

name

Name of the content filtering action to be modified.

serviceName

Service to which to forward HTTP requests. Required if the qualifier is FORWARD.

value

String containing the header_name and header_value. If the qualifier is ADD, specify <header_name>: <header_value>. If the qualifier is CORRUPT, specify only the header_name

respCode

Response code to be returned for HTTP requests (for use with the ERRORCODE qualifier).

Minimum value: 1

page

HTML page to return for HTTP requests (For use with the ERRORCODE qualifier).

Example

```
set filter action bad_url_action -respcode 400 -page "<HTML>Bad URL.</HTML>" set filter a
```

unset filter action

Use this command to remove filter action settings. Refer to the set filter action command for meanings of the arguments.

Synopsys

unset filter action <name> -page

show filter action

Displays information about available filtering actions.

Synopsys

show filter action [<name>]

Arguments

name

Name of the content filtering action to be displayed. If a name is not provided, information about all filter actions is shown.

Outputs

qual

Qualifier, which is the action to be performed. The qualifier cannot be changed after it is set. The available options function as follows:

ADD - Adds the specified HTTP header.

RESET - Terminates the connection, sending the appropriate termination notice to the user's browser.

FORWARD - Redirects the request to the designated service. You must specify either a service name or a page, but not both.

DROP - Silently deletes the request, without sending a response to the user's browser.

CORRUPT - Modifies the designated HTTP header to prevent it from performing the function it was intended to perform, then sends the request/response to the server/browser.

ERRORCODE. Returns the designated HTTP error code to the user's browser (for example, 404, the standard HTTP code for a non-existent Web page).

serviceName

The service to which HTTP requests are forwarded. This parameter will exist when the qualifier is FORWARD.

value

The string containing the header_name and header_value. When the qualifier is ADD it will have header_name:header_value. When the qualifier is Corrupt this will have header_name.

respCode

The response code to be returned for HTTP requests. This parameter will exist when the qualifier is ERRORCODE.

page

The HTML page that will be returned for the HTTP requests. This parameter will exist when the qualifier is ERRORCODE.

stateflag

isDefault

A value of true is returned if it is a default filteraction.

flag

builtin

devno

count

Example

Example 1 The following shows an example of the output of the show filter action command v

filter global

The following operations can be performed on "filter global":

[bind](#) | [unbind](#) | [show](#)

bind filter global

Apply (bind) the specified filtering policy globally. Note: Filtering requires the content filtering license.

Synopsis

```
bind filter global (<policyName> [-priority <positive_integer>]) [-state ( ENABLED | DISABLED )]
```

Arguments

policyName

Name of the filtering policy to be bound.

priority

Priority assigned to the policy.

Minimum value: 0

Maximum value: 64000

state

State of the binding.

Possible values: ENABLED, DISABLED

Default value: ENABLED

Example

```
To send RESET for all the HTTP requests which are not get or head type, following filter ;
```

unbind filter global

Deactivate a globally bound filter policy.

Synopsis

```
unbind filter global <policyName>
```

Arguments

policyName

Name of the filter policy to be unbound.

Example

```
Globally active filter policies can be seen using command: show filter global 1) Pol:
```

show filter global

Displays the globally activated filter policies.

Synopsys

show filter global

Outputs

policyName

The name of the filter policy.

priority

The priority of the policy.

state

State of the binding.

stateflag

devno

count

Example

```
show filter global 1)          Policy Name: url_filter Priority: 0 2)          Policy Name: reset_
```


filter htmlinjectionparameter

The following operations can be performed on "filter htmlinjectionparameter":

[set](#) | [unset](#) | [show](#)

set filter htmlinjectionparameter

Sets the HTML injection parameters.

Synopsis

```
set filter htmlinjectionparameter [-rate <positive_integer>] [-frequency <positive_integer>] [-strict ( ENABLED | DISABLED )] [-htmlsearchlen <positive_integer>]
```

Arguments

rate

For a rate of x, HTML injection is done for 1 out of x policy matches.

Default value: 1

Minimum value: 1

frequency

For a frequency of x, HTML injection is done at least once per x milliseconds.

Default value: 1

Minimum value: 1

strict

Searching for <html> tag. If this parameter is enabled, HTML injection does not insert the prebody or postbody content unless the <html> tag is found.

Possible values: ENABLED, DISABLED

Default value: ENABLED

htmlsearchlen

Number of characters, in the HTTP body, in which to search for the <html> tag if strict mode is set.

Default value: 1024

Minimum value: 1

Example

```
set htmlinjection parameter -rate 10 -frequency 1
```

unset filter htmlinjectionparameter

Removes the HTML injection settings..Refer to the set filter htmlinjectionparameter command for meanings of the arguments.

Synopsis

```
unset filter htmlinjectionparameter [-rate] [-frequency] [-strict] [-htmlsearchlen]
```

Example

a) unset htmlinjectionparameter -rate b) unset htmlinjectionparameter -frequency c) un:

show filter htmlinjectionparameter

Displays the HTML injection parameters.

Synopsys

show filter htmlinjectionparameter

Outputs

rate

For a rate of x, HTML injection is done for 1 out of x policy matches.

frequency

For a frequency of x, HTML injection is done at least once per x milliseconds.

strict

Searching for <html> tag. If this parameter is enabled, HTML injection does not insert the prebody or postbody content unless the <html> tag is found.

htmlsearchlen

Number of characters, in the HTTP body, in which to search for the <html> tag if strict mode is set.

Example

rate : 10

filter htmlinjectionvariable

The following operations can be performed on "filter htmlinjectionvariable":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add filter htmlinjectionvariable

Creates an HTML injection variable.

Synopsis

```
add filter htmlinjectionvariable <variable> [-value <string>]
```

Arguments

variable

Name for the HTML injection variable to be added.

value

Value to be assigned to the new variable.

Example

```
add htmlinjectionvariable EDGESIGHT_SERVER_IP -value 1.1.1.1
```

rm filter htmlinjectionvariable

Removes an HTML injection variable.

Synopsis

```
rm filter htmlinjectionvariable <variable>
```

Arguments

variable

Name of the HTML injection variable to be removed.

Example

```
rm htmlinjectionvariable EDGESIGHT_SERVER_IP
```

set filter htmlinjectionvariable

Modifies the value of an HTML injection variable.

Synopsis

```
set filter htmlinjectionvariable <variable> [-value <string>]
```

Arguments

variable

Name of the HTML injection variable to be modified.

value

Value to be assigned to the new variable.

Example

```
set htmlinjectionvariable EDGESIGHT_SERVER_IP -value 2.2.2.2
```

unset filter htmlinjectionvariable

Use this command to remove filter htmlinjectionvariable settings. Refer to the set filter htmlinjectionvariable command for meanings of the arguments.

Synopsys

```
unset filter htmlinjectionvariable <variable> -value
```

show filter htmlinjectionvariable

Displays information about HTML injection variables.

Synopsys

```
show filter htmlinjectionvariable [<variable>]
```

Arguments

variable

Name of the HTML injection variable to be displayed. If a name is not provided, information about all the HTML injection variables is shown.

Outputs

value

Value of the HTML injection variable

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

type

Type of the HTML injection variable

devno

count

stateflag

Example

```
show htmlinjectionvariable EDGESIGHT_SERVER_IP
```

filter policy

The following operations can be performed on "filter policy":

[add](#) | [rm](#) | [set](#) | [show](#)

add filter policy

Creates a content filtering policy.

Synopsys

```
add filter policy <name> -rule <expression> (-reqAction <string> | -resAction <string>)
```

Arguments

name

Name for the filtering action. Must begin with a letter, number, or the underscore character (_). Other characters allowed, after the first character, are the hyphen (-), period (.), pound (#), space (), at (@), equals (=), and colon (:) characters. Choose a name that helps identify the type of action. The name cannot be updated after the policy is created.

CLI Users: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my policy" or 'my policy').

rule

NetScaler classic expression specifying the type of connections that match this policy.

reqAction

Name of the action to be performed on requests that match the policy. Cannot be specified if the rule includes condition to be evaluated for responses.

resAction

The action to be performed on the response. The string value can be a filter action created filter action or a built-in action.

Example

```
Example 1: add policy expression e1 "sourceip == 66.33.22.0 -netmask 255.255.255.0" add p
```

rm filter policy

Removes a filter policy.

Synopsys

```
rm filter policy <name>
```

Arguments

name

Name of the filter policy to be removed.

Example

```
rm filter policy filter_policy_name The "show filter policy" command shows all filter pol:
```

set filter policy

Modifies a filter policy.

Synopsys

```
set filter policy <name> [-rule <expression>] [-reqAction <string> | -resAction <string>]
```

Arguments

name

Name of the filter policy to be modified.

rule

NetScaler classic expression specifying the type of connections that match this policy.

reqAction

Name of the action to be performed on requests that match the policy. Cannot be specified if the rule includes condition to be evaluated for responses.

resAction

The action to be performed on the response. The string value can be a filter action created filter action or a built-in action.

Example

Example 1: A filter policy to allow access of URL /foo/secure.asp only from 65.186.55.0 n

show filter policy

Displays information about the filter policies.

Synopsys

```
show filter policy [<name>]
```

Arguments

name

Name of the filter policy to be displayed. If a name is not provided, information about all the filter policies is shown.

Outputs

rule

NetScaler classic expression specifying the type of connections that match this policy.

reqAction

The name of the action to be performed on the request.

resAction

The action to be performed on the response.

hits

boundTo

The entity name to which policy is bound

activePolicy

priority

bindPolicyType

policyType

devno

count

stateflag

Example

```
show filter policy 1)      Name: nimda_filter Rule: (URL CONTAINS root.exe || URL CONTAIN:
```

filter postbodyInjection

The following operations can be performed on "filter postbodyInjection":

[set](#) | [unset](#) | [show](#)

set filter postbodyInjection

Specifies the file to be used for postbody injection.

Synopsys

```
set filter postbodyInjection <postbody>
```

Arguments

postbody

Name of file whose contents are to be inserted after the response body.

Example

```
set filter postbodyInjection ens/postbody.js
```

unset filter postbodyInjection

Removes the setting that specifies the file used for postbody injection..Refer to the set filter postbodyInjection command for meanings of the arguments.

Synopsys

```
unset filter postbodyInjection [-postbody]
```

Example

```
unset filter postbodyInjection
```

show filter postbodyInjection

Displays the name of the file used for postbody injection.

Synopsys

```
show filter postbodyInjection
```

Outputs

postbody

The name of the postbody file.

systemIID

The system IID of the current NetScaler system.

filter prebodyInjection

The following operations can be performed on "filter prebodyInjection":

[set](#) | [unset](#) | [show](#)

set filter prebodyInjection

Specifies the file to be used for prebody injection.

Synopsys

```
set filter prebodyInjection <prebody>
```

Arguments

prebody

Name of file whose contents are to be inserted before the response body.

Example

```
set filter prebodyInjection ens/prebody.js
```

unset filter prebodyInjection

Removes the setting that specifies the file used for prebody injection..Refer to the set filter prebodyInjection command for meanings of the arguments.

Synopsys

```
unset filter prebodyInjection [-prebody]
```

Example

```
unset filter prebodyInjection
```

show filter prebodyInjection

Displays the name of the file used for prebody injection.

Synopsys

```
show filter prebodyInjection
```

Outputs

prebody

The name of the prebody file.

systemIID

The system IID of the current NetScaler system.

GSLB Commands

The entities on which you can perform NetScaler CLI operations:

- o `gslb action`
- o `gslb config`
- o `gslb domain`
- o `gslb Idnsentries`
- o `gslb Idnsentry`
- o `gslb parameter`
- o `gslb policy`
- o `gslb runningConfig`
- o `gslb service`
- o `gslb site`
- o `gslb syncStatus`
- o `gslb vserver`

gslb action

The following operations can be performed on "gslb action":

`add` | `rm` | `set` | `show`

add gslb action

Add GSLB action used in the GSLB policy NOTE: This command is deprecated.

Synopsys

Arguments

name

The name of the GSLB action

preferredLocation

The target site to be returned in the DNS response when a policy is successfully evaluated against the incoming DNS request. Target site is specified in dotted notation with up to 6 qualifiers. Wildcard '*' is accepted as a valid qualifier token.

Example

```
add gslb action pref_site -preferredlocation NorthAmerica.US.*.*.*.*
```

rm gslb action

Remove the gslb action configured in the system NOTE: This command is deprecated.

Synopsys

Arguments

name

The name of the action to be removed

Example

```
rm gslb action redirect_asia
```

set gslb action

Change the preferredlocation of the given gslb action NOTE: This command is deprecated.

Synopsys

Arguments

name

The name of the GSLB action

preferredLocation

The target site to be returned in the DNS response when a policy is successfully evaluated against the incoming DNS request. Target site is specified in dotted notation with up to 6 qualifiers. Wildcard '*' is accepted as a valid qualifier token.

Example

```
set gslb action pref_site -preferredlocation NorthAmerica.US.*.*.*.*
```

show gslb action

Display the GSLB actions configured NOTE: This command is deprecated.

Synopsys

Arguments

name

The name of the action.

Outputs

preferredLocation

The target site to be returned in the DNS response when a policy is successfully evaluated against the incoming DNS request. Target site is specified in dotted notation with up to 6 qualifiers. Wildcard '*' is accepted as a valid qualifier token.

devno

count

stateflag

Example

```
show gslb action
```

gslb config

The following operations can be performed on "gslb config":

sync gslb config

Synchronizes the GSLB running configuration on all NetScaler appliances participating in the GSLB setup. The appliance on which this command is run is considered the master node. All GSLB sites configured on the master node and not having a parent site are synchronized with the master node.

Synopsys

```
sync gslb config [-preview | -forceSync <string> | -command <string> | -nowarn | -saveconfig] [-debug]
```

Arguments

preview

Do not synchronize the GSLB sites, but display the commands that would be applied on the slave node upon synchronization. Mutually exclusive with the Save Configuration option.

debug

Generate verbose output when synchronizing the GSLB sites. The Debug option generates more verbose output than the sync gslb config command in which the option is not used, and is useful for analyzing synchronization issues.

forceSync

Force synchronization of the specified site even if a dependent configuration on the remote site is preventing synchronization or if one or more GSLB entities on the remote site have the same name but are of a different type. You can specify either the name of the remote site that you want to synchronize with the local site, or you can specify All Sites in the configuration utility (the string all-sites in the CLI). If you specify All Sites, all the sites in the GSLB setup are synchronized with the site on the master node.

Note: If you select the Force Sync option, the synchronization starts without displaying the commands that are going to be executed.

nowarn

Suppress the warning and the confirmation prompt that are displayed before site synchronization begins. This option can be used in automation scripts that must not be interrupted by a prompt.

saveconfig

Save the configuration on all the nodes participating in the synchronization process, automatically. The master saves its configuration immediately before synchronization begins. Slave nodes save their configurations after the process of synchronization is complete. A slave node saves its configuration only if the configuration difference was successfully applied to it. Mutually exclusive with the Preview option.

command

Run the specified command on the master node and then on all the slave nodes. You cannot use this option with the force sync and preview options.

Example

```
sync gslb config
```

gslb domain

The following operations can be performed on "gslb domain":

[show](#) | [stat](#)

show gslb domain

Displays the bounded attributes of the domain.

Synopsys

show gslb domain [<name>]

Arguments

name

Name of the Domain

Outputs

serviceType

The type GSLB service

state

The state of the vserver

dnsRecordType

The IP type for this GSLB vserver.

stateChangeTimeSec

Time since last state change

lbMethod

The load balancing method set for the virtual server

backupLBMethod

Indicates the backup method in case the primary fails

persistenceType

Indicates if persistence is set on the gslb vserver

persistenceId

Persistence id of the gslb vserver

serviceName

The service name.

vServerName

siteName

Name of the site to which the service belongs.

cip

Indicates if Client IP option is enabled

sitePersistence

Indicates the type of cookie persistence set

sitePrefix

The site prefix string.

gslbthreshold

The threshold value of the service

httpRequest

HTTP request to the backend server

ipTunnel

The state of the monitor for tunneled devices.

customHeaders

The string that is sent to the service. Applicable to HTTP ,HTTP-ECV and RTSP monitor types.

respCode

The response codes.

monitorName

Monitor name

netmask

Netmask

v6netmasklen

Number of bits to consider, in an IPv6 source IP address, for creating the hash that is required by the SOURCEIPHASH load balancing method.

EDR

Send clients an empty DNS response when the GSLB virtual server is DOWN.

MIR**dynamicWeight**

Dynamic weight method of the vserver

persistMask

The optional IPv4 network mask applied to IPv4 addresses to establish source IP address based persistence.

v6persistmasklen

Number of bits to consider in an IPv6 source IP address when creating source IP address based persistence sessions.

IPAddress

The Ip address of the service

port

Port Number

weight

weight assigned

dynamicConfWt

dynamic weight

cumulativeWeight

cumulative weight

svrEffGslbState

GSLB server state

cnameEntry

The cname of the gslb service

monState

Monitor state

monitorTotalProbes

Total monitor probes

monitorTotalFailedProbes

Total probes failed

monitorCurrentFailedProbes

Total number of current failed probes

monStatCode

The code indicating the monitor response.

monStatParam1

First parameter for use with message code.

monStatParam2

Second parameter for use with message code.

monStatParam3

Third parameter for use with message code.

responseTime

Response time of this monitor.

lastresponse

The string form of monstatcode.

stateflag

stateflag

devno**count**

stat gslb domain

Displays the statistics associated with a global server load balancing (GSLB) domain.

Synopsys


```
stat gslb domain [<name> [-dnsRecordType <dnsRecordType>]] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

name

Name of the GSLB domain for which to display statistics. If you do not specify a name, statistics are shown for all configured GSLB

domains.

dnsRecordType

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

domainHits (Hits)

Total number of DNS queries received.

Dns Record Type (Rec_Type)

Type of DNS record returned

gslb ldnentries

The following operations can be performed on "gslb ldnentries":

[clear](#) | [show](#)

clear gslb ldnentries

Clears all the local DNS (LDNS) entries created on the NetScaler appliance. LDNS entries store network metrics for RTT learned from the packets exchanged with LDNS servers.

Synopsys

```
clear gslb ldnentries
```

show gslb ldnentries

Displays the local DNS (LDNS) entries created on the NetScaler appliance. LDNS entries store network metrics for RTT learned from the packets exchanged with LDNS servers.

Synopsys

```
show gslb ldnentries
```

Outputs

siteName

The GSLB site name

numsites

Specifies the number of gslb sites

IPAddress

IP address of the LDNS server

TTL

TTL value of the LDNS entry

name

Monitor that is currently being used to monitor the LDNS ip..

rtt

RTT value of the LDNS entry for all gslb sites

devno

count

stateflag

Example

```
show gslb ldnentries
```

gslb ldnsentry

The following operations can be performed on "gslb ldnsentry":

rm gslb ldnsentry

Removes the LDNS entry for the specified LDNS IP address.

Synopsis

```
rm gslb ldnsentry <IPAddress>
```

Arguments

IPAddress

IP address of the LDNS server.

Example

```
rm gslb ldnsentry 10.102.27.226
```

gslb parameter

The following operations can be performed on "gslb parameter":

[set](#) | [unset](#) | [show](#)

set gslb parameter

Sets various global GSLB parameters.

Synopsys

```
set gslb parameter [-ldnsEntryTimeout <secs>] [-RTTTolerance <msecs>] [-ldnsMask <netmask>] [-v6ldnsmasklen <positive_integer>] [-ldnsProbeOrder <ldnsProbeOrder> ...] [-dropLdnsReq ( ENABLED | DISABLED )]
```

Arguments

ldnsEntryTimeout

Time, in seconds, after which an inactive LDNS entry is removed.

Default value: 180

Minimum value: 30

Maximum value: 65534

RTTTolerance

Tolerance, in milliseconds, for newly learned round-trip time (RTT) values. If the difference between the old RTT value and the newly computed RTT value is less than or equal to the specified tolerance value, the LDNS entry in the network metric table is not updated with the new RTT value. Prevents the exchange of metrics when variations in RTT values are negligible.

Default value: 5

Minimum value: 1

Maximum value: 100

ldnsMask

The IPv4 network mask with which to create LDNS entries.

Default value: 0xFFFFFFFF

v6ldnsmasklen

Mask for creating LDNS entries for IPv6 source addresses. The mask is defined as the number of leading bits to consider, in the source IP address, when creating an LDNS entry.

Default value: 128

Minimum value: 1

Maximum value: 128

ldnsProbeOrder

Order in which monitors should be initiated to calculate RTT.

Possible values: PING, DNS, TCP

Default value: ARRAY(0x2b03df50)

dropLdnsReq

Drop LDNS requests if round-trip time (RTT) information is not available.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
set gslb parameter -ldnsMask 255.255.0.0
```

unset gslb parameter

Use this command to remove gslb parameter settings. Refer to the set gslb parameter command for meanings of the arguments.

Synopsys

```
unset gslb parameter [-ldnsEntryTimeout] [-RTTTolerance] [-ldnsMask] [-v6ldnsmasklen] [-ldnsProbeOrder] [-dropLdnsReq]
```

show gslb parameter

Displays the global GSLB parameters.

Synopsys

```
show gslb parameter
```

Outputs

flags

State of the GSLB parameter.

ldnsEntryTimeout

Time, in seconds, after which an inactive LDNS entry is removed.

RTTTolerance

Tolerance, in milliseconds, for newly learned round-trip time (RTT) values. If the difference between the old RTT value and the newly computed RTT value is less than or equal to the specified tolerance value, the LDNS entry in the network metric table is not updated with the new RTT value. Prevents the exchange of metrics when variations in RTT values are negligible.

ldnsMask

The IPv4 network mask with which to create LDNS entries.

v6ldnsmasklen

Mask for creating LDNS entries for IPv6 source addresses. The mask is defined as the number of leading bits to consider, in the source IP address, when creating an LDNS entry.

ldnsProbeOrder

The order in which monitors should be initiated to calculate RTT

dropLdnsReq

Drop LDNS requests if round-trip time (RTT) information is not available.

Example

```
show gslb parameter
```

gslb policy

The following operations can be performed on "gslb policy":

add | **rm** | **set** | **show**

add gslb policy

Add GSLB policy NOTE: This command is deprecated.

Synopsys

Arguments

name

The name of the GSLB policy

reqRule

The expression rule

action

The GSLB action to be used when the reqrule is matched

Example

```
add gslb policy gslb_redirect -reqRule client_Japan -action pref_site
```

rm gslb policy

Remove the gslb policy configured in the system NOTE: This command is deprecated.

Synopsys

Arguments

name

The name of the policy to be removed

Example

```
rm gslb policy gslb_redirect
```

set gslb policy

Change the action for the given gslb policy NOTE: This command is deprecated.

Synopsys

Arguments

name

The name of the gslb policy.

action

The action to be taken for the given gslb policy

Example

```
set gslb policy gslb_redirect -action redirect_asia
```

show gslb policy

Display the configured GSLB policy NOTE: This command is deprecated.

Synopsys

Arguments

name

The name of the GSLB policy.

Outputs

reqRule

The expression rule

action

The action taken for the given gslb policy.

hits

Number of policy hits for the gslb policy.

devno

count

stateflag

Example

```
show gslb policy
```

gslb runningConfig

The following operations can be performed on "gslb runningConfig":

show gslb runningConfig

Displays the complete GSLB configuration running on the NetScaler appliance. In addition to the saved configuration, the running configuration includes GSLB settings that have not yet been saved to the NetScaler configuration file (ns.conf).

Synopsys

show gslb runningConfig

Outputs

response

gslb sync status as text blob

gslb service

The following operations can be performed on "gslb service":

add | **rm** | **set** | **unset** | **bind** | **unbind** | **show** | **stat** | **rename**

add gslb service

Creates a global server load balancing (GSLB) service.

Synopsis

```
add gslb service <serviceName> (-cnameEntry <string> | <IP> | <serverName> | <serviceType> | <port> | -publicIP <ip_addr|ipv6_addr|*> | -publicPort <port> | -sitePersistence <sitePersistence> | -sitePrefix <string>) [-maxClient <positive_integer>] [-healthMonitor ( YES | NO )] [-siteName <string> [-state ( ENABLED | DISABLED )] [-cip ( ENABLED | DISABLED ) [-<cipHeader>]] [-cookieTimeout <mins>] [-cltTimeout <secs>] [-svrTimeout <secs>] [-maxBandwidth <positive_integer>] [-downStateFlush ( ENABLED | DISABLED )] [-maxAAAUUsers <positive_integer>] [-monThreshold <positive_integer>] [-hashId <positive_integer>] [-comment <string>] [-appflowLog ( ENABLED | DISABLED )]
```

Arguments

serviceName

Name for the GSLB service. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Can be changed after the GSLB service is created.

CLI Users: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my gslbsvc" or 'my gslbsvc').

cnameEntry

Canonical name of the GSLB service. Used in CNAME-based GSLB.

IP

IP address for the GSLB service. Should represent a load balancing, content switching, or VPN virtual server on the NetScaler appliance, or the IP address of another load balancing device.

serverName

Name of the server hosting the GSLB service.

serviceType

Type of service to create.

Possible values: HTTP, FTP, TCP, UDP, SSL, SSL_BRIDGE, SSL_TCP, NNTP, ANY, SIP_UDP, RADIUS, RDP, RTSP, MYSQL, MSSQL, ORACLE

Default value: NSSVC_SERVICE_UNKNOWN

port

Port on which the load balancing entity represented by this GSLB service listens.

Minimum value: 1

publicIP

The public IP address that a NAT device translates to the GSLB service's private IP address. Optional.

publicPort

The public port associated with the GSLB service's public IP address. The port is mapped to the service's private port number. Applicable to the local GSLB service. Optional.

maxClient

The maximum number of open connections that the service can support at any given time. A GSLB service whose connection count reaches the maximum is not considered when a GSLB decision is made, until the connection count drops below the maximum.

Minimum value: 0

Maximum value: 4294967294

healthMonitor

Monitor the health of the GSLB service.

Possible values: YES, NO

Default value: YES

siteName

Name of the GSLB site to which the service belongs.

state

Enable or disable the service.

Possible values: ENABLED, DISABLED

Default value: ENABLED

cip

In the request that is forwarded to the GSLB service, insert a header that stores the client's IP address. Client IP header insertion is used in connection-proxy based site persistence.

Possible values: ENABLED, DISABLED

Default value: DISABLED

cipHeader

Name for the HTTP header that stores the client's IP address. Used with the Client IP option. If client IP header insertion is enabled on the service and a name is not specified for the header, the NetScaler appliance uses the name specified by the cipHeader parameter in the set ns param command or, in the GUI, the Client IP Header parameter in the Configure HTTP Parameters dialog box.

sitePersistence

Use cookie-based site persistence. Applicable only to HTTP and SSL GSLB services.

Possible values: ConnectionProxy, HTTPRedirect, NONE

cookieTimeout

Timeout value, in minutes, for the cookie, when cookie based site persistence is enabled.

Maximum value: 1440

sitePrefix

The site's prefix string. When the service is bound to a GSLB virtual server, a GSLB site domain is generated internally for each bound service-domain pair by concatenating the site prefix of the service and the name of the domain. If the special string NONE is specified, the site-prefix string is unset. When implementing HTTP redirect site persistence, the NetScaler appliance redirects GSLB requests to GSLB services by using their site domains.

cltTimeout

Idle time, in seconds, after which a client connection is terminated. Applicable if connection proxy based site persistence is used.

Maximum value: 31536000

svrTimeout

Idle time, in seconds, after which a server connection is terminated. Applicable if connection proxy based site persistence is used.

Maximum value: 31536000

maxBandwidth

Integer specifying the maximum bandwidth allowed for the service. A GSLB service whose bandwidth reaches the maximum is not considered when a GSLB decision is made, until its bandwidth consumption drops below the maximum.

Minimum value: 0

downStateFlush

Flush all active transactions associated with the GSLB service when its state transitions from UP to DOWN. Do not enable this option for services that must complete their transactions. Applicable if connection proxy based site persistence is used.

Possible values: ENABLED, DISABLED

maxAAUsers

Maximum number of SSL VPN users that can be logged on concurrently to the VPN virtual server that is represented by this GSLB service. A GSLB service whose user count reaches the maximum is not considered when a GSLB decision is made, until the count drops below the maximum.

Minimum value: 0

Maximum value: 65535

monThreshold

Monitoring threshold value for the GSLB service. If the sum of the weights of the monitors that are bound to this GSLB service and are in the UP state is not equal to or greater than this threshold value, the service is marked as DOWN.

Minimum value: 0

Maximum value: 65535

hashId

Unique hash identifier for the GSLB service, used by hash based load balancing methods.

Minimum value: 1

comment

Any comments that you might want to associate with the GSLB service.

appflowLog

Enable logging appflow flow information

Possible values: ENABLED, DISABLED

Default value: ENABLED

Example

```
add gslb service sj_svc 203.12.123.12 http 80 -site san_jos
```

rm gslb service

Removes a global server load balancing (GSLB) service configured on the appliance.

Synopsis

```
rm gslb service <serviceName>
```

Arguments

serviceName

Name of the GSLB service.

Example

```
rm gslb service sj_svc
```

set gslb service

Modifies the specified parameters of a global server load balancing (GSLB) service.

Synopsis

```
set gslb service <serviceName> [-IPAddress <ip_addr|ipv6_addr|*>] [-publicIP <ip_addr|ipv6_addr|*>] [-publicPort <port>] [-cip ( ENABLED | DISABLED ) [<cipHeader>]] [-sitePersistence <sitePersistence>] [-sitePrefix <string>] [-maxClient <positive_integer>] [-healthMonitor ( YES | NO )] [-maxBandwidth <positive_integer>] [-downStateFlush ( ENABLED | DISABLED )] [-maxAAAUUsers <positive_integer>] [-viewName <string> <viewIP>] [-monThreshold <positive_integer>] [-weight <positive_integer> <monitorName>] [-hashId <positive_integer>] [-comment <string>] [-appflowLog ( ENABLED | DISABLED )]
```

Arguments

serviceName

Name of the GSLB service.

IPAddress

The new IP address of the service.

publicIP

The public IP address that a NAT device translates to the GSLB service's private IP address. Optional.

publicPort

The public port associated with the GSLB service's public IP address. The port is mapped to the service's private port number. Applicable to the local GSLB service. Optional.

Minimum value: 1

cip

In the request that is forwarded to the GSLB service, insert a header that stores the client's IP address. Client IP header insertion is used in connection-proxy based site persistence.

Possible values: ENABLED, DISABLED

Default value: DISABLED

cipHeader

Name for the HTTP header that stores the client's IP address. Used with the Client IP option. If client IP header insertion is enabled on the service and a name is not specified for the header, the NetScaler appliance uses the name specified by the cipHeader parameter in the set ns param command or, in the GUI, the Client IP Header parameter in the Configure HTTP Parameters dialog box.

sitePersistence

Use cookie-based site persistence. Applicable only to HTTP and SSL GSLB services.

Possible values: ConnectionProxy, HTTPRedirect, NONE

sitePrefix

The site's prefix string. When the service is bound to a GSLB virtual server, a GSLB site domain is generated internally for each bound service-domain pair by concatenating the site prefix of the service and the name of the domain. If the special string NONE is specified, the site-prefix string is unset. When implementing HTTP redirect site persistence, the NetScaler appliance redirects GSLB requests to GSLB services by using their site domains.

maxClient

The maximum number of open connections that the service can support at any given time. A GSLB service whose connection count reaches the maximum is not considered when a GSLB decision is made, until the connection count drops below the maximum.

Minimum value: 0

Maximum value: 4294967294

healthMonitor

Monitor the health of the GSLB service.

Possible values: YES, NO

Default value: YES

maxBandwidth

Maximum bandwidth.

Minimum value: 0

downStateFlush

Flush all active transactions associated with the GSLB service when its state transitions from UP to DOWN. Do not enable this option for services that must complete their transactions. Applicable if connection proxy based site persistence is used.

Possible values: ENABLED, DISABLED

Default value: ENABLED

maxAAUsers

Maximum number of SSL VPN users that can be logged on concurrently to the VPN virtual server that is represented by this GSLB service. A GSLB service whose user count reaches the maximum is not considered when a GSLB decision is made, until the count drops below the maximum.

Minimum value: 0

Maximum value: 65535

viewName

Name of the DNS view of the service. A DNS view is used in global server load balancing (GSLB) to return a predetermined IP address to a specific group of clients, which are identified by using a DNS policy.

viewIP

IP address to be used for the given view

monThreshold

Monitoring threshold value for the GSLB service. If the sum of the weights of the monitors that are bound to this GSLB service and are in the UP state is not equal to or greater than this threshold value, the service is marked as DOWN.

Minimum value: 0

Maximum value: 65535

weight

Weight to assign to the monitor-service binding. A larger number specifies a greater weight. Contributes to the monitoring threshold, which determines the state of the service.

Minimum value: 1

Maximum value: 100

monitorName

Name of the monitor to bind to the service.

hashId

Unique hash identifier for the GSLB service, used by hash based load balancing methods.

Minimum value: 1

comment

Any comments that you might want to associate with the GSLB service.

appflowLog

Enable logging appflow flow information

Possible values: ENABLED, DISABLED

Default value: ENABLED

Example

```
set gslb service sj_svc -sitePersistence ConnectionProxy
```

unset gslb service

Use this command to remove gslb service settings. Refer to the set gslb service command for meanings of the arguments.

Synopsis

```
unset gslb service <serviceName> [-publicIP] [-publicPort] [-cip] [-cipHeader] [-sitePersistence] [-sitePrefix] [-maxClient] [-healthMonitor] [-maxBandwidth] [-downStateFlush] [-maxAAUsers] [-monThreshold] [-hashId] [-comment] [-appflowLog]
```

bind gslb service

Binds a DNS view or a monitor to a global server load balancing (GSLB) service.

Synopsis

```
bind gslb service <serviceName> ((-viewName <string> <viewIP>) | (-monitorName <string>@ [-monState ( ENABLED | DISABLED )] [-weight <positive_integer>]))
```

Arguments

serviceName

Name of the GSLB service.

viewName

Name of the DNS view of the service. A DNS view is used in global server load balancing (GSLB) to return a predetermined IP address to a specific group of clients, which are identified by using a DNS policy.

viewIP

IP address for the specified DNS view.

monitorName

Name of the monitor to bind to the GSLB service.

monState

Initial state of the GSLB monitor.

Possible values: ENABLED, DISABLED

Default value: ENABLED

weight

Weight to assign to the monitor-service binding. A larger number specifies a greater weight. Contributes to the monitoring threshold, which determines the state of the service.

Default value: 1

Minimum value: 1

Maximum value: 100

Example

```
bind gslb service -viewName privateview 1.2.3.4
```

unbind gslb service

Unbinds a DNS view or a monitor from a global server load balancing (GSLB) service.

Synopsys

```
unbind gslb service <serviceName> (-viewName <string> | -monitorName <string>@)
```

Arguments

serviceName

Name of the GSLB service.

viewName

Name of the DNS view of the service. A DNS view specifies the IP address that must be returned to clients accessing the service from a specific location.

monitorName

Name of the monitor to unbind.

Example

```
unbind gslb service -viewName privateview
```

show gslb service

Displays the parameters of all the global server load balancing (GSLB) services configured on the appliance, or the parameters of just the specified service, and statistics related to the service. To display the parameters of all the GSLB services, do not specify a service name.

Synopsys

show gslb service [<serviceName>] show gslb service stats - alias for 'stat gslb service'

Arguments

serviceName

Name of the GSLB service.

Outputs

gslb

IPAddress

IP address of the service

IP

IP address of the service

serverName

Name of the server hosting the GSLB service.

serviceType

Service type.

port

Port number of the service.

publicIP

Public ip of the service

publicPort

Public port of the service

maxClient

Maximum number of clients.

maxAAAUUsers

Maximum number of SSL VPN users that can be logged on concurrently to the VPN virtual server that is represented by this GSLB service. A GSLB service whose user count reaches the maximum is not considered when a GSLB decision is made, until the count drops below the maximum.

siteName

Name of the site to which the service belongs.

svrState

Server state.

svrEffGslbState

Effective state of the gslb svc

gslbthreshold

Indicates if gslb svc has reached threshold

gslbsvcStats

Indicates if gslb svc has stats of the primary or the whole chain

state

Enable or disable the service.

monitorName

Monitor name.

monState

The running state of the monitor on this service.

cip

Indicates if Client IP option is enabled

cipHeader

The client IP header used in the HTTP request.

sitePersistence

Indicates the type of cookie persistence set

sitePrefix

The site prefix string.

cltTimeout

Client timeout in seconds.

svrTimeout

Server timeout in seconds.

totalfailedprobes

The total number of failed probs.

preferredLocation

Preferred location.

maxBandwidth

Maximum bandwidth.

downStateFlush

Flush all active transactions associated with the GSLB service when its state transitions from UP to DOWN. Do not enable this option for services that must complete their transactions. Applicable if connection proxy based site persistence is used.

cnameEntry

Canonical name of the GSLB service. Used in CNAME-based GSLB.

viewName

Name of the DNS view of the service. A DNS view is used in global server load balancing (GSLB) to return a predetermined IP address to a specific group of clients, which are identified by using a DNS policy.

viewIP

IP address to be used for the given view

weight

The Weight of monitor

monThreshold

Monitoring threshold value for the GSLB service. If the sum of the weights of the monitors that are bound to this GSLB service and are in the UP state is not equal to or greater than this threshold value, the service is marked as DOWN.

failedprobes

Number of the current failed monitoring probes.

monStatCode

The code indicating the monitor response.

monStatParam1

First parameter for use with message code.

monStatParam2

Second parameter for use with message code.

monStatParam3

Third parameter for use with message code.

responseTime

Response time of this monitor.

hashId

Unique hash identifier for the GSLB service, used by hash based load balancing methods.

comment

Any comments that you might want to associate with the GSLB service.

stateflag

stateflag

healthMonitor

Monitor the health of the GSLB service.

appflowLog

Enable logging appflow flow information

svccfgFlags

Contains the information about config info like internal/configured service

monitorTotalProbes

Total number of probes sent to monitor this service.

monitorTotalFailedProbes

Total number of failed probes

monitorCurrentFailedProbes

Total number of currently failed probes

stateChangeTimeSec

Time when last state change happened. Seconds part.

ticksSinceLastStateChange

Time in 10 millisecond ticks since the last state change.

threshold**CIMonOwner**

Tells the mon owner of the gslb service.

CIMonView

Tells the view id of the monitoring owner.

devno**count**

Example

```
show gslb service sj_svc
```

stat gslb service

Displays the statistical data collected for a global server load balancing (GSLB) service.

Synopsys

```
stat gslb service [<serviceName>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

serviceName

Name of the GSLB service.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Current Client Est connections (CIntEstConn)

Number of client connections in ESTABLISHED state.

IP address (IP)

The IP address on which the service is running.

Port (port)

The port on which the service is running.

Service type (Type)

The service type of this service. Possible values are ADNS, DNS, MYSQL, RTSP, SSL_DIAMETER, ADNS_TCP, DNS_TCP, NNTP, SIP_UDP, SSL_TCP, ANY, FTP, RADIUS, SNMP, TCP, DHCPRA, HTTP, RDP, SSL, TFTP, DIAMETER, MSSQL, RPCSVR, SSL_BRIDGE, UDP

State

Current state of the server. Possible values are UP, DOWN, UNKNOWN, OFS(Out of Service), TROFS (Transition Out of Service), TROFS_DOWN(Down When going Out of Service)

Request bytes (Reqb)

Total number of request bytes received on this service or virtual server.

Response bytes (Rspb)

Number of response bytes received by this service or virtual server.

Current load on the service (Load)

Load on the service that is calculated from the bound load based monitor.

Requests (Req)

Total number of requests received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

Responses (Rsp)

Number of responses received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

Current client connections (CIntConn)

Number of current client connections.

Current server connections (SvrConn)

Number of current connections to the actual servers behind the virtual server.

Service hits (Hits)

Number of times that the service has been provided.

Current Server Est connections (SvrEstConn)

Number of server connections in ESTABLISHED state.

rename gslb service

Renames a global server load balancing (GSLB) service.

Synopsys

```
rename gslb service <serviceName>@ <newName>@
```

Arguments

serviceName

Existing name of the GSLB service.

newName

New name for the GSLB service.

Example

```
rename gslb service gsl_svc gslb_svc_new
```

gslb site

The following operations can be performed on "gslb site":

add | **rm** | **set** | **unset** | **show** | **stat**

add gslb site

Creates a global server load balancing site.

Synopsys

```
add gslb site <siteName> [<siteType>] <siteIPAddress> [-publicIP <ip_addr|ipv6_addr|*>] [-metricExchange (
ENABLED | DISABLED )] [-nwMetricExchange ( ENABLED | DISABLED )] [-sessionExchange ( ENABLED |
DISABLED )] [-triggerMonitor <triggerMonitor>] [-parentSite <string>] [-clip <ip_addr|ipv6_addr|*> [<publicCLIP>]]
```

Arguments

siteName

Name for the GSLB site. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the virtual server is created.

CLI Users: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my gslbsite" or 'my gslbsite').

siteType

Type of site to create. If the type is not specified, the appliance automatically detects and sets the type on the basis of the IP address being assigned to the site. If the specified site IP address is owned by the appliance (for example, a MIP address or SNIP address), the site is a local site. Otherwise, it is a remote site.

Possible values: REMOTE, LOCAL

Default value: NONE

siteIPAddress

IP address for the GSLB site. The GSLB site uses this IP address to communicate with other GSLB sites. For a local site, use any IP address that is owned by the appliance (for example, a SNIP or MIP address, or the IP address of the ADNS service).

publicIP

Public IP address for the local site. Required only if the appliance is deployed in a private address space and the site has a public IP address hosted on an external firewall or a NAT device.

metricExchange

Exchange metrics with other sites. Metrics are exchanged by using Metric Exchange Protocol (MEP). The appliances in the GSLB setup exchange health information once every second.

If you disable metrics exchange, you can use only static load balancing methods (such as round robin, static proximity, or the hash-based methods), and if you disable metrics exchange when a dynamic load balancing method (such as least connection) is in operation, the appliance falls back to round robin. Also, if you disable metrics exchange, you must use a monitor to determine the state of GSLB services. Otherwise, the service is marked as DOWN.

Possible values: ENABLED, DISABLED

Default value: ENABLED

nwMetricExchange

Exchange, with other GSLB sites, network metrics such as round-trip time (RTT), learned from communications with various local DNS (LDNS) servers used by clients. RTT information is used in the dynamic RTT load balancing method, and is exchanged every 5 seconds.

Possible values: ENABLED, DISABLED

Default value: ENABLED

sessionExchange

Exchange persistent session entries with other GSLB sites every five seconds.

Possible values: ENABLED, DISABLED

Default value: ENABLED

triggerMonitor

Specify the conditions under which the GSLB service must be monitored by a monitor, if one is bound.
Available settings function as follows:

* ALWAYS - Monitor the GSLB service at all times.

* MEPDOWN - Monitor the GSLB service only when the exchange of metrics through the Metrics Exchange Protocol (MEP) is disabled.

MEPDOWN_SVCDOWN - Monitor the service in either of the following situations:

* The exchange of metrics through MEP is disabled.

* The exchange of metrics through MEP is enabled but the status of the service, learned through metrics exchange, is DOWN.

Possible values: ALWAYS, MEPDOWN, MEPDOWN_SVCDOWN

Default value: ALWAYS

parentSite

Parent site of the GSLB site, in a parent-child topology.

clip

Cluster IP used to connect to remote cluster site for GSLB autosync

publicCLIP

Public cluster IP used to connect to remote cluster site for GSLB autosync if the remote cluster is behind a NAT

Example

```
add site new_york LOCAL 192.168.100.12 -publicIP 65.200.211.139
```

rm gslb site

Removes a global server load balancing (GSLB) site and all its constituent GSLB services.

Synopsys

```
rm gslb site <siteName>
```

Arguments

siteName

Name of the GSLB site to remove.

Example

```
rm gslb site new_york
```

set gslb site

Modifies the specified parameters of a global server load balancing (GSLB) site.

Synopsys

```
set gslb site <siteName> [-metricExchange ( ENABLED | DISABLED )] [-nwMetricExchange ( ENABLED | DISABLED )] [-sessionExchange ( ENABLED | DISABLED )] [-triggerMonitor <triggerMonitor>]
```

Arguments

siteName

Name of the GSLB site.

metricExchange

Exchange metrics with other sites. Metrics are exchanged by using Metric Exchange Protocol (MEP). The appliances in the GSLB setup exchange health information once every second.

If you disable metrics exchange, you can use only static load balancing methods (such as round robin, static proximity, or the hash-based methods), and if you disable metrics exchange when a dynamic load balancing method (such as least connection) is in operation, the appliance falls back to round robin. Also, if you disable metrics exchange, you must use a monitor to determine the state of GSLB services. Otherwise, the service is marked as DOWN.

Possible values: ENABLED, DISABLED

Default value: ENABLED

nwMetricExchange

Exchange, with other GSLB sites, network metrics such as round-trip time (RTT), learned from communications with various local DNS (LDNS) servers used by clients. RTT information is used in the dynamic RTT load balancing method, and is exchanged every 5 seconds.

Possible values: ENABLED, DISABLED

Default value: ENABLED

sessionExchange

Exchange persistent session entries with other GSLB sites every five seconds.

Possible values: ENABLED, DISABLED

Default value: ENABLED

triggerMonitor

Specify the conditions under which the GSLB service must be monitored by a monitor, if one is bound. Available settings function as follows:

* ALWAYS - Monitor the GSLB service at all times.

* MEPDOWN - Monitor the GSLB service only when the exchange of metrics through the Metrics Exchange Protocol (MEP) is disabled.

MEPDOWN_SVCDOWN - Monitor the service in either of the following situations:

* The exchange of metrics through MEP is disabled.

* The exchange of metrics through MEP is enabled but the status of the service, learned through metrics exchange, is DOWN.

Possible values: ALWAYS, MEPDOWN, MEPDOWN_SVCDOWN

Default value: ALWAYS

Example

```
set gslb site new_york - metricExchange DISABLED
```

unset gslb site

Use this command to remove gslb site settings. Refer to the set gslb site command for meanings of the arguments.

Synopsys

```
unset gslb site <siteName> [-metricExchange] [-nwMetricExchange] [-sessionExchange] [-triggerMonitor]
```

show gslb site

Displays the parameters of all the GSLB sites configured on the appliance, or the parameters of the specified GSLB site.

Synopsys

```
show gslb site [<siteName>] show gslb site stats - alias for 'stat gslb site'
```

Arguments

siteName

Name of the GSLB site. If you specify a site name, details of all the site's constituent services are also displayed.

Outputs

siteType

Specifies whether the site is LOCAL or REMOTE.

siteIPAddress

The IP address of the site.

publicIP

The Public IP of the gslb site.

metricExchange

Exchange metrics with other sites. Metrics are exchanged by using Metric Exchange Protocol (MEP). The appliances in the GSLB setup exchange health information once every second.

If you disable metrics exchange, you can use only static load balancing methods (such as round robin, static proximity, or the hash-based methods), and if you disable metrics exchange when a dynamic load balancing method (such as least connection) is in operation, the appliance falls back to round robin. Also, if you disable metrics exchange, you must use a monitor to determine the state of GSLB services. Otherwise, the service is marked as DOWN.

serviceName

Service name.

IPAddress

IP Address of the gslb service.

port

Port number of the gslb service.

state

State of the gslb service.

status

Current metric exchange status.

persistenceMEPStatus

Network metric and persistence exchange MEP connection status

serviceType

Service type.

nwMetricExchange

Specifies whether the exchange of network metrics like RTT is enabled or disabled.

sessionExchange

Specifies whether the exchange of persistence session entries is enabled or disabled.

triggerMonitor

Specify the conditions under which the GSLB service must be monitored by a monitor, if one is bound.
Available settings function as follows:

* ALWAYS - Monitor the GSLB service at all times.

* MEPDOWN - Monitor the GSLB service only when the exchange of metrics through the Metrics Exchange Protocol (MEP) is disabled.

MEPDOWN_SVCDOWN - Monitor the service in either of the following situations:

* The exchange of metrics through MEP is disabled.

* The exchange of metrics through MEP is enabled but the status of the service, learned through metrics exchange, is DOWN.

parentSite

Parent site of the GSLB site, in a parent-child topology.

cnameEntry

The cname of the gslb service.

stateflag

stateflag

version

will be true if the remote site's version is ncore compatible with the local site.(>= 9.2)

clip

Cluster IP used to connect to remote cluster site for GSLB autosync

publicCLIP

Public cluster IP used to connect to remote cluster site for GSLB autosync if the remote cluster is behind a NAT

devno

count

Example

```
show site new_york
```

stat gslb site

Displays statistics for a GSLB site.

Synopsys

```
stat gslb site [<siteName>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full )]
```

Arguments

siteName

Name of the GSLB site for which to display detailed statistics. If a name is not specified, basic information about all GSLB sites is displayed.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Gslb Site Public IP address (Public IP)

The public IP address of this GSLB site.

Gslb Site private IP address (Private IP)

The private IP address of this GSLB site.

Metric Exchange State (MEPstate)

Indicates the status of the Metric Exchange Policy at this GSLB site.

Persistence Exchange (PersMEP)

Indicates whether Persistence entries exchange is enabled or disabled at this GSLB site.

Network Metric Exchange (NwMEP)

Indicates whether network metric exchange is enabled or disabled at this GSLB site.

Metric Exchange (MEP)

Indicates whether metric exchange is enabled or disabled at this GSLB site.

GSLB Site type (sitetype)

Indicates whether this GSLB site is local or remote.

Gslb Site public IP address (Public IP)

The public IP address of this GSLB site.

Site Metric Metric Exchange State (SiteMetricMEPstate)

Indicates the status of the site metric Metric Exchange connection at this GSLB site.

Netowrk Metric Metric Exchange State (NwMetricMEPstate)

Indicates the status of the network metric Metric Exchange connection at this GSLB site.

Request bytes (Reqb)

Total number of request bytes received by the virtual servers represented by all GSLB services associated with this GSLB site.

Response bytes (Rspb)

Number of response bytes received by the virtual servers represented by all GSLB services associated with this GSLB site.

Requests (Req)

Total number of requests received by the virtual servers represented by all GSLB services associated with this GSLB site.

Responses (Rsp)

Number of responses received by the virtual servers represented by all GSLB services associated with this GSLB site.

Current client connections (CIntConn)

Number of current client connections to the virtual servers represented by all GSLB services associated with this GSLB site.

Current server connections (SvrConn)

Number of current connections to the real servers behind the virtual servers represented by all GSLB services associated with this GSLB site.

gslb syncStatus

The following operations can be performed on "gslb syncStatus":

show gslb syncStatus

Displays the status of the last GSLB configuration synchronization.

Synopsys

show gslb syncStatus

Outputs

response

gslb sync status as text blob

gslb vserver

The following operations can be performed on "gslb vserver":

add | **rm** | **set** | **unset** | **bind** | **unbind** | **enable** | **disable** | **show** | **stat** | **rename**

add gslb vserver

Creates a global server load balancing (GSLB) virtual server.

Synopsis

```
add gslb vserver <name> <serviceType> [-dnsRecordType <dnsRecordType>] [-lbMethod <lbMethod>] [-
backupLBMethod <backupLBMethod>] [-netmask <netmask>] [-v6netmasklen <positive_integer>] [-tolerance
<positive_integer>] [-persistenceType ( SOURCEIP | NONE )] [-persistenceld <positive_integer>] [-persistMask
<netmask>] [-v6persistmasklen <positive_integer>] [-timeout <mins>] [-EDR ( ENABLED | DISABLED )] [-MIR (
ENABLED | DISABLED )] [-disablePrimaryOnDown ( ENABLED | DISABLED )] [-dynamicWeight <dynamicWeight>]
[-state ( ENABLED | DISABLED )] [-considerEffectiveState ( NONE | STATE_ONLY )] [-comment <string>] [-
soMethod <soMethod>] [-soPersistence ( ENABLED | DISABLED )] [-soPersistenceTimeOut <positive_integer>] [-
soThreshold <positive_integer>] [-soBackupAction <soBackupAction>] [-appflowLog ( ENABLED | DISABLED )]
```

Arguments

name

Name for the GSLB virtual server. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Can be changed after the virtual server is created.

CLI Users:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my vserver" or 'my vserver').

serviceType

Protocol used by services bound to the virtual server.

Possible values: HTTP, FTP, TCP, UDP, SSL, SSL_BRIDGE, SSL_TCP, NNTP, ANY, SIP_UDP, RADIUS, RDP, RTSP, MYSQL, MSSQL, ORACLE

dnsRecordType

DNS record type to associate with the GSLB virtual server's domain name.

Possible values: A, AAAA, CNAME

Default value: A

lbMethod

Load balancing method for the GSLB virtual server.

Possible values: ROUNDROBIN, LEASTCONNECTION, LEASTRESPONSETIME, SOURCEIPHASH, LEASTBANDWIDTH, LEASTPACKETS, STATICPROXIMITY, RTT, CUSTOMLOAD

Default value: LEASTCONNECTION

backupLBMethod

Backup load balancing method. Becomes operational if the primary load balancing method fails or cannot be used. Valid only if the primary method is based on either round-trip time (RTT) or static proximity.

Possible values: ROUNDROBIN, LEASTCONNECTION, LEASTRESPONSETIME, SOURCEIPHASH, LEASTBANDWIDTH, LEASTPACKETS, STATICPROXIMITY, RTT, CUSTOMLOAD

netmask

IPv4 network mask for use in the SOURCEIPHASH load balancing method.

Default value: 0xFFFFFFFF

v6netmasklen

Number of bits to consider, in an IPv6 source IP address, for creating the hash that is required by the SOURCEIPHASH load balancing method.

Default value: 128

Minimum value: 1

Maximum value: 128

tolerance

Site selection tolerance, in milliseconds, for implementing the RTT load balancing method. If a site's RTT deviates from the lowest RTT by more than the specified tolerance, the site is not considered when the NetScaler appliance makes a GSLB decision. The appliance implements the round robin method of global server load balancing between sites whose RTT values are within the specified tolerance. If the tolerance is 0 (zero), the appliance always sends clients the IP address of the site with the lowest RTT.

Minimum value: 0

Maximum value: 100

persistenceType

Use source IP address based persistence for the virtual server.

After the load balancing method selects a service for the first packet, the IP address received in response to the DNS query is used for subsequent requests from the same client.

Possible values: SOURCEIP, NONE

persistenceId

The persistence ID for the GSLB virtual server. The ID is a positive integer that enables GSLB sites to identify the GSLB virtual server, and is required if source IP address based or spill over based persistence is enabled on the virtual server.

Minimum value: 0

Maximum value: 65535

persistMask

The optional IPv4 network mask applied to IPv4 addresses to establish source IP address based persistence.

Default value: 0xFFFFFFFF

v6persistmasklen

Number of bits to consider in an IPv6 source IP address when creating source IP address based persistence sessions.

Default value: 128

Minimum value: 1

Maximum value: 128

timeout

Idle time, in minutes, after which a persistence entry is cleared.

Default value: 2

Minimum value: 2

Maximum value: 1440

EDR

Send clients an empty DNS response when the GSLB virtual server is DOWN.

Possible values: ENABLED, DISABLED

Default value: DISABLED

MIR

Include multiple IP addresses in the DNS responses sent to clients.

Possible values: ENABLED, DISABLED

Default value: DISABLED

disablePrimaryOnDown

Continue to direct traffic to the backup chain even after the primary GSLB virtual server returns to the UP state. Used when spillover is configured for the virtual server.

Possible values: ENABLED, DISABLED

Default value: DISABLED

dynamicWeight

Specify if the appliance should consider the service count, service weights, or ignore both when using weight-based load balancing methods. The state of the number of services bound to the virtual server help the appliance to select the service.

Possible values: SERVICECOUNT, SERVICEWEIGHT, DISABLED

Default value: DISABLED

state

State of the GSLB virtual server.

Possible values: ENABLED, DISABLED

Default value: ENABLED

considerEffectiveState

If the primary state of all bound GSLB services is DOWN, consider the effective states of all the GSLB services, obtained through the Metrics Exchange Protocol (MEP), when determining the state of the GSLB virtual server. To consider the effective state, set the parameter to STATE_ONLY. To disregard the effective state, set the parameter to NONE.

The effective state of a GSLB service is the ability of the corresponding virtual server to serve traffic. The effective state of the load balancing virtual server, which is transferred to the GSLB service, is UP even if only one virtual server in the backup chain of virtual servers is in the UP state.

Possible values: NONE, STATE_ONLY

Default value: NONE

comment

Any comments that you might want to associate with the GSLB virtual server.

soMethod

Type of threshold that, when exceeded, triggers spillover. Available settings function as follows:

* CONNECTION - Spillover occurs when the number of client connections exceeds the threshold.

* DYNAMICCONNECTION - Spillover occurs when the number of client connections at the GSLB virtual server exceeds the sum of the maximum client (Max Clients) settings for bound GSLB services. Do not specify a spillover threshold for this setting, because the threshold is implied by the Max Clients settings of the bound GSLB services.

* BANDWIDTH - Spillover occurs when the bandwidth consumed by the GSLB virtual server's incoming and outgoing traffic exceeds the threshold.

* HEALTH - Spillover occurs when the percentage of weights of the GSLB services that are UP drops below the threshold. For example, if services gslbSvc1, gslbSvc2, and gslbSvc3 are bound to a virtual server, with weights 1, 2, and 3, and the spillover threshold is 50%, spillover occurs if gslbSvc1 and gslbSvc3 or gslbSvc2 and gslbSvc3 transition to DOWN.

* NONE - Spillover does not occur.

Possible values: CONNECTION, DYNAMICCONNECTION, BANDWIDTH, HEALTH, NONE

soPersistence

If spillover occurs, maintain source IP address based persistence for both primary and backup GSLB virtual servers.

Possible values: ENABLED, DISABLED

Default value: DISABLED

soPersistenceTimeout

Timeout for spillover persistence, in minutes.

Default value: 2

Minimum value: 2

Maximum value: 1440

soThreshold

Threshold at which spillover occurs. Specify an integer for the CONNECTION spillover method, a bandwidth value in kilobits per second for the BANDWIDTH method (do not enter the units), or a percentage for the HEALTH method (do not enter the percentage symbol).

Minimum value: 1

Maximum value: 4294967287

soBackupAction

Action to be performed if spillover is to take effect, but no backup chain to spillover is usable or exists

Possible values: DROP, ACCEPT, REDIRECT

appflowLog

Enable logging appflow flow information

Possible values: ENABLED, DISABLED

Default value: ENABLED

Example

```
add gslb vserver gvip http
```

rm gslb vserver

Removes a global server load balancing (GSLB) virtual server configured on the appliance.

Synopsys

rm gslb vserver <name>

Arguments

name

Name of the GSLB virtual server to remove.

Example

```
rm gslb vserver gvip
```

set gslb vserver

Modifies the specified parameters of a global server load balancing (GSLB) virtual server.

Synopsys

```
set gslb vserver <name> [-dnsRecordType <dnsRecordType>] [-backupVServer <string>] [-lbMethod <lbMethod>] [-
backupLBMethod <backupLBMethod>] [-netmask <netmask>] [-v6netmasklen <positive_integer>] [-tolerance
<positive_integer>] [-persistenceType ( SOURCEIP | NONE )] [-persistenceld <positive_integer>] [-persistMask
<netmask>] [-v6persistmasklen <positive_integer>] [-timeout <mins>] [-EDR ( ENABLED | DISABLED )] [-MIR (
ENABLED | DISABLED )] [-disablePrimaryOnDown ( ENABLED | DISABLED )] [-dynamicWeight <dynamicWeight>]
[-considerEffectiveState ( NONE | STATE_ONLY )] [-soMethod <soMethod>] [-soPersistence ( ENABLED |
DISABLED )] [-soPersistenceTimeOut <positive_integer>] [-soThreshold <positive_integer>] [-soBackupAction
<soBackupAction>] [-serviceName <string> -weight <positive_integer>] [-domainName <string> [-TTL <secs>] [-
backupIP <ip_addr|ipv6_addr|*>] [-cookieDomain <string>] [-cookieTimeout <mins>] [-sitedomainTTL <secs>]] [-
comment <string>] [-appflowLog ( ENABLED | DISABLED )]
```

Arguments

name

Name of the GSLB virtual server.

dnsRecordType

DNS record type to associate with the GSLB virtual server's domain name.

Possible values: A, AAAA, CNAME

Default value: A

backupVServer

Name of the backup GSLB virtual server to which the appliance should forward requests if the status of the primary GSLB virtual server is down or exceeds its spillover threshold.

lbMethod

Load balancing method for the GSLB virtual server.

Possible values: ROUNDROBIN, LEASTCONNECTION, LEASTRESPONSETIME, SOURCEIPHASH, LEASTBANDWIDTH, LEASTPACKETS, STATICPROXIMITY, RTT, CUSTOMLOAD

Default value: LEASTCONNECTION

backupLBMethod

Backup load balancing method. Becomes operational if the primary load balancing method fails or cannot be used. Valid only if the primary method is based on either round-trip time (RTT) or static proximity.

Possible values: ROUNDROBIN, LEASTCONNECTION, LEASTRESPONSETIME, SOURCEIPHASH, LEASTBANDWIDTH, LEASTPACKETS, STATICPROXIMITY, RTT, CUSTOMLOAD

netmask

IPv4 network mask for use in the SOURCEIPHASH load balancing method.

Default value: 0xFFFFFFFF

v6netmasklen

Number of bits to consider, in an IPv6 source IP address, for creating the hash that is required by the SOURCEIPHASH load balancing method.

Default value: 128

Minimum value: 1

Maximum value: 128

tolerance

Site selection tolerance, in milliseconds, for implementing the RTT load balancing method. If a site's RTT deviates from the lowest RTT by more than the specified tolerance, the site is not considered when the NetScaler appliance makes a GSLB decision. The appliance implements the round robin method of global server load balancing between sites whose RTT values are within the specified tolerance. If the tolerance is 0 (zero), the appliance always sends clients the IP address of the site with the lowest RTT.

Minimum value: 0

Maximum value: 100

persistenceType

Persistence type for the virtual server. Possible value for this parameter is SOURCEIP, which specifies persistence based on the source IP address of inbound packets. After the load balancing method selects a link for transmission of the first packet, the IP address received in response to the DNS query is used for subsequent requests from the same client.

Possible values: SOURCEIP, NONE

persistenceId

The persistence ID for the GSLB virtual server. The ID is a positive integer that enables GSLB sites to identify the GSLB virtual server, and is required if source IP address based or spill over based persistence is enabled on the virtual server.

Minimum value: 0

Maximum value: 65535

persistMask

The optional IPv4 network mask applied to IPv4 addresses to establish source IP address based persistence.

Default value: 0xFFFFFFFF

v6persistmasklen

Number of bits to consider in an IPv6 source IP address when creating source IP address based persistence sessions.

Default value: 128

Minimum value: 1

Maximum value: 128

timeout

Idle time, in minutes, after which a persistence entry is cleared.

Default value: 2

Minimum value: 2

Maximum value: 1440

EDR

Send clients an empty DNS response when the GSLB virtual server is DOWN.

Possible values: ENABLED, DISABLED

Default value: DISABLED

MIR

Include multiple IP addresses in the DNS responses sent to clients.

Possible values: ENABLED, DISABLED

Default value: DISABLED

disablePrimaryOnDown

Continue to direct traffic to the backup chain even after the primary GSLB virtual server returns to the UP state. Used when spillover is configured for the virtual server.

Possible values: ENABLED, DISABLED

Default value: DISABLED

dynamicWeight

Specify if the appliance should consider the service count, service weights, or ignore both when using weight-based load balancing methods. The state of the number of services bound to the virtual server help the appliance to select the service.

Possible values: SERVICECOUNT, SERVICEWEIGHT, DISABLED

Default value: DISABLED

considerEffectiveState

If the primary state of all bound GSLB services is DOWN, consider the effective states of all the GSLB services, obtained through the Metrics Exchange Protocol (MEP), when determining the state of the GSLB virtual server. To consider the effective state, set the parameter to STATE_ONLY. To disregard the effective state, set the parameter to NONE.

The effective state of a GSLB service is the ability of the corresponding virtual server to serve traffic. The effective state of the load balancing virtual server, which is transferred to the GSLB service, is UP even if only one virtual server in the backup chain of virtual servers is in the UP state.

Possible values: NONE, STATE_ONLY

Default value: NONE

soMethod

Type of threshold that, when exceeded, triggers spillover. Available settings function as follows:

* CONNECTION - Spillover occurs when the number of client connections exceeds the threshold.

* DYNAMICCONNECTION - Spillover occurs when the number of client connections at the GSLB virtual server exceeds the sum of the maximum client (Max Clients) settings for bound GSLB services. Do not specify a spillover threshold for this setting, because the threshold is implied by the Max Clients settings of the bound GSLB services.

* BANDWIDTH - Spillover occurs when the bandwidth consumed by the GSLB virtual server's incoming and outgoing traffic exceeds the threshold.

* HEALTH - Spillover occurs when the percentage of weights of the GSLB services that are UP drops below the threshold. For example, if services gslbSvc1, gslbSvc2, and gslbSvc3 are bound to a virtual server, with weights 1, 2, and 3, and the spillover threshold is 50%, spillover occurs if gslbSvc1 and gslbSvc3 or gslbSvc2 and gslbSvc3 transition to DOWN.

* NONE - Spillover does not occur.

Possible values: CONNECTION, DYNAMICCONNECTION, BANDWIDTH, HEALTH, NONE

soPersistence

If spillover occurs, maintain source IP address based persistence for both primary and backup GSLB virtual servers.

Possible values: ENABLED, DISABLED

Default value: DISABLED

soPersistenceTimeout

Timeout for spillover persistence, in minutes.

Default value: 2

Minimum value: 2

Maximum value: 1440

soThreshold

Threshold at which spillover occurs. Specify an integer for the CONNECTION spillover method, a bandwidth value in kilobits per second for the BANDWIDTH method (do not enter the units), or a percentage for the HEALTH method (do not enter the percentage symbol).

Minimum value: 1

Maximum value: 4294967287

soBackupAction

Action to be performed if spillover is to take effect, but no backup chain to spillover is usable or exists

Possible values: DROP, ACCEPT, REDIRECT

serviceName

Name of the GSLB service for which to change the weight.

weight

Weight to assign to the GSLB service.

Minimum value: 1

Maximum value: 100

domainName

Domain name for which to change the time to live (TTL) and/or backup service IP address.

TTL

Time to live (TTL) for the domain.

Minimum value: 1

backupIP

The IP address of the backup service for the specified domain name. Used when all the services bound to the domain are down, or when the backup chain of virtual servers is down.

cookieDomain

The cookie domain for the GSLB site. Used when inserting the GSLB site cookie in the HTTP response.

cookieTimeout

Timeout, in minutes, for the GSLB site cookie.

Maximum value: 1440

sitedomainTTL

TTL, in seconds, for all internally created site domains (created when a site prefix is configured on a GSLB service) that are associated with this virtual server.

Minimum value: 1

comment

Any comments that you might want to associate with the GSLB virtual server.

appflowLog

Enable logging appflow flow information

Possible values: ENABLED, DISABLED

Default value: ENABLED

Example

```
set gslb vserver gvip -persistenceType SOURCEIP
```

unset gslb vserver

Removes the specified settings from the specified global server load balancing (GSLB) virtual server. Attributes for which a default value is available revert to their default values..Refer to the set gslb vserver command for meanings of the arguments.

Synopsys

```
unset gslb vserver <name>@ [-backupVServer] [-dnsRecordType] [-lbMethod] [-backupLBMethod] [-netmask] [-v6netmasklen] [-tolerance] [-persistenceType] [-persistenceld] [-persistMask] [-v6persistmasklen] [-timeout] [-EDR] [-MIR] [-disablePrimaryOnDown] [-dynamicWeight] [-considerEffectiveState] [-soMethod] [-soPersistence] [-soPersistenceTimeOut] [-soBackupAction] [-serviceName] [-weight] [-comment] [-appflowLog]
```

Example

```
unset gslb vserver lb_vip -backupVServer For multiple gslb vservers the command is: unset
```

bind gslb vserver

Binds a domain, service, backup IP address, or cookie domain to a GSLB virtual server.

Synopsys

```
bind gslb vserver <name> ((-serviceName <string> [-weight <positive_integer>]) | (-domainName <string> [-TTL <secs>] [-backupIP <ip_addr|ipv6_addr|*>] [-cookieDomain <string>] [-cookieTimeout <mins>] [-sitedomainTTL <secs>]) | (-policyName <string>@ [-priority <positive_integer>] [-gotoPriorityExpression <expression>] [-type (REQUEST | RESPONSE )]))
```

Arguments

name

Name of the virtual server on which to perform the binding operation.

serviceName

Name of the GSLB service for which to change the weight.

weight

Weight to assign to the GSLB service.

Default value: 1

Minimum value: 1

Maximum value: 100

domainName

Domain name for which to change the time to live (TTL) and/or backup service IP address.

TTL

Time to live (TTL) for the domain.

Minimum value: 1

backupIP

The IP address of the backup service for the specified domain name. Used when all the services bound to the domain are down, or when the backup chain of virtual servers is down.

cookieDomain

The cookie domain for the GSLB site. Used when inserting the GSLB site cookie in the HTTP response.

cookieTimeout

Timeout, in minutes, for the GSLB site cookie.

sitedomainTTL

TTL, in seconds, for all internally created site domains (created when a site prefix is configured on a GSLB service) that are associated with this virtual server.

Default value: 3600

Minimum value: 1

policyName

Name of the policy bound to the GSLB vserver.

priority

Priority.

Minimum value: 1

Maximum value: 2147483647

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

o If gotoPriorityExpression is not present or if it is equal to END then the policy bank evaluation ends here

o Else if the gotoPriorityExpression is equal to NEXT then the next policy in the priority order is evaluated.

o Else gotoPriorityExpression is evaluated. The result of gotoPriorityExpression (which has to be a number) is processed as follows:

- An UNDEF event is triggered if
 - . gotoPriorityExpression cannot be evaluated
 - . gotoPriorityExpression evaluates to number which is smaller than the maximum priority in the policy bank but is not same as any policy's priority
 - . gotoPriorityExpression evaluates to a priority that is smaller than the current policy's priority
- If the gotoPriorityExpression evaluates to the priority of the current policy then the next policy in the priority order is evaluated.
- If the gotoPriorityExpression evaluates to the priority of a policy further ahead in the list then that policy will be evaluated next.

This field is applicable only to rewrite and responder policies.

type

Bind point to which to bind the policy.

Possible values: REQUEST, RESPONSE

Example

```
bind gslb vserver gvip -domainName www.mynw.com
```

unbind gslb vserver

Unbinds the domain or service from the GSLB virtual server.

Synopsys

```
unbind gslb vserver <name> (-serviceName <string> | (-domainName <string> [-backupIP] [-cookieDomain]) | -policyName <string>@)
```

Arguments

name

Name of the GSLB virtual server.

serviceName

Name of the GSLB service for which to change the weight.

domainName

Domain name for which to change the time to live (TTL) and/or backup service IP address.

backupIP

The IP address of the backup service for the specified domain name. Used when all the services bound to the domain are down, or when the backup chain of virtual servers is down.

cookieDomain

The cookie domain for the GSLB site. Used when inserting the GSLB site cookie in the HTTP response.

policyName

The policy that has been bound to this load balancing virtual server, using the `###bind gslb vserver###` command.

Example

```
unbind gslb vserver gvip -domainName www.mynw.com
```


enable gslb vserver

Enables a global server load balancing (GSLB) virtual server that has been disabled. (A GSLB virtual server is enabled by default.)

Synopsis

```
enable gslb vserver <name>@
```

Arguments

name

Name of the GSLB virtual server to enable.

Example

```
enable gslb vserver gslb_vip To enable multiple gslb vservers use the following command:
```

disable gslb vserver

Disables a global server load balancing (GSLB) virtual server and takes it out of service.

Synopsis

```
disable gslb vserver <name>@
```

Arguments

name

Name of the GSLB virtual server to disable.

Example

```
disable gslb vserver gslb_vip To disable multiple gslb vservers use the following command:
```

show gslb vserver

Displays the parameters of all the global server load balancing (GSLB) virtual servers configured on the appliance, or the parameters of the specified GSLB virtual server.

Synopsis

```
show gslb vserver [<name>] show gslb vserver stats - alias for 'stat gslb vserver'
```

Arguments

name

Name of the GSLB virtual server.

Outputs

serviceType

Protocol used by services bound to the virtual server.

state

State of the gslb vserver.

ipType

The IP type for this GSLB vserver.

dnsRecordType

The IP type for this GSLB vserver.

persistenceType

Indicates if persistence is set on the gslb vserver

persistenceId

Persistence id of the gslb vserver

lbMethod

The load balancing method set for the virtual server

backupLBMethod

Indicates the backup method in case the primary fails

tolerance

Indicates the deviation we can tolerate when we have the LB method as RTT

timeout

Idle timeout for persistence entries.

netmask

The netmask used in the SOURCEIPHASH policy.

v6netmasklen

The netmask used for ipv6 traffic in the SOURCE/DEST IPHASH policy.

persistMask

The netmask used while SOURCEIP based persistency is ENABLED.

v6persistmasklen

The netmask applied for ipv6 traffic when the persistency type is SOURCEIP.

serviceName

The service name.

weight

Weight for the service.

domainName

The name of the domain for which TTL and/or backupIP has changed.

TTL

TTL for the given domain.

backupIP

Backup IP for the given domain.

cookieDomain

The cookie domain for the GSLB domain. This will be used when inserting the GSLB site cookie in the HTTP response. By default, cookie domain will not be inserted.

cookieTimeout

Time out value of the cookie in minutes

sitedomainTTL

Site domain TTL.

IPAddress

IP address.

port

Port number.

status

Current status of the gslb vserver. During the initial phase if the configured lb method is not round robin , the vserver will adopt round robin to distribute traffic for a predefined number of requests.

lbrreason

Reason why a vserver is in RR. The following are the reasons:

- 1 - MEP is DOWN (GSLB)
- 2 - LB method has changed
- 3 - Bound service's state changed to UP
- 4 - A new service is bound
- 5 - Startup RR factor has changed
- 6 - LB feature is enabled
- 7 - Load monitor is not active on a service
- 8 - Vserver is Enabled
- 9 - SSL feature is Enabled
- 10 - All bound services have reached threshold. Using effective state to load balance (GSLB)
- 11 - Primary state of bound services are not UP. Using effective state to load balance (GSLB)
- 12 - No LB decision can be made as all bound services have either reached threshold or are not UP (GSLB)
- 13 - All load monitors are active

preferredLocation

The target site to be returned in the DNS response when a policy is successfully evaluated against the incoming DNS request. Target site is specified in dotted notation with up to 6 qualifiers. Wildcard '*' is accepted as a valid qualifier token.

backupVServer

Backup vserver in case the primary fails

backupSessionTimeout

A non zero value enables the feature. The minimum value is 2 minutes. To disable the feature set the value to zero. The created session is in effect for a specific client per domain.

EDR

Indicates if Empty Down Response is enabled/disabled

MIR

Indicates if Multi IP Response is enabled/disabled

disablePrimaryOnDown

Continue to direct traffic to the backup chain even after the primary GSLB virtual server returns to the UP state. Used when spillover is configured for the virtual server.

dynamicWeight

Dynamic weight method. Possible values are, the svc count or the svc weights or ignore both.

isCname

is cname feature set on vserver

cumulativeWeight

Cumulative weight is the weight of GSLB service considering both its configured weight and dynamic weight. It is equal to product of dynamic weight and configured weight of the gslb service

dynamicConfWt

Weight obtained by the virtue of bound service count or weight

thresholdValue

Tells whether threshold exceeded for this service participating in CUSTOMLB

sitePersistence

Type of Site Persistence set

svrEffGslbState

Effective state of the gslb svc

gslbthreshold

Indicates if gslb svc has reached threshold

considerEffectiveState

If the primary state of all bound GSLB services is DOWN, consider the effective states of all the GSLB services, obtained through the Metrics Exchange Protocol (MEP), when determining the state of the GSLB virtual server. To consider the effective state, set the parameter to STATE_ONLY. To disregard the effective state, set the parameter to NONE.

The effective state of a GSLB service is the ability of the corresponding virtual server to serve traffic. The effective state of the load balancing virtual server, which is transferred to the GSLB service, is UP even if only one virtual server in the backup chain of virtual servers is in the UP state.

cnameEntry

The cname of the gslb service.

totalServices

Total number of services bound to the vserver.

activeServices

Total number of active services bound to the vserver.

stateChangeTimeSec

Time when last state change happened. Seconds part.

stateChangeTimemSec

Time at which last state change happened. Milliseconds part.

ticksSinceLastStateChange

Time in 10 millisecond ticks since the last state change.

comment

Any comments that you might want to associate with the GSLB virtual server.

soPersistenceTimeOut

Timeout for spillover persistence, in minutes.

soMethod

Type of threshold that, when exceeded, triggers spillover. Available settings function as follows:

- * CONNECTION - Spillover occurs when the number of client connections exceeds the threshold.
- * DYNAMICCONNECTION - Spillover occurs when the number of client connections at the GSLB virtual server exceeds the sum of the maximum client (Max Clients) settings for bound GSLB services. Do not specify a spillover threshold for this setting, because the threshold is implied by the Max Clients settings of the bound GSLB services.
- * BANDWIDTH - Spillover occurs when the bandwidth consumed by the GSLB virtual server's incoming and outgoing traffic exceeds the threshold.
- * HEALTH - Spillover occurs when the percentage of weights of the GSLB services that are UP drops below the threshold. For example, if services gslbSvc1, gslbSvc2, and gslbSvc3 are bound to a virtual server, with weights 1, 2, and 3, and the spillover threshold is 50%, spillover occurs if gslbSvc1 and gslbSvc3 or gslbSvc2 and gslbSvc3 transition to DOWN.
- * NONE - Spillover does not occur.

soBackupAction

Action to be performed if spillover is to take effect, but no backup chain to spillover is usable or exists

soPersistence

If spillover occurs, maintain source IP address based persistence for both primary and backup GSLB virtual servers.

soThreshold

Threshold at which spillover occurs. Specify an integer for the CONNECTION spillover method, a bandwidth value in kilobits per second for the BANDWIDTH method (do not enter the units), or a percentage for the HEALTH method (do not enter the percentage symbol).

health

Health of vserver based on percentage of weights of active svcs/all svcs. This does not consider administratively disabled svcs

stateflag

stateflag

appflowLog

Enable logging appflow flow information

policyName

Name of the policy bound to the GSLB vserver.

priority

Priority.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

- o If gotoPriorityExpression is not present or if it is equal to END then the policy bank evaluation ends here

- o Else if the gotoPriorityExpression is equal to NEXT then the next policy in the priority order is evaluated.

- o Else gotoPriorityExpression is evaluated. The result of gotoPriorityExpression (which has to be a number) is processed as follows:

- An UNDEF event is triggered if

- . gotoPriorityExpression cannot be evaluated

- . gotoPriorityExpression evaluates to number which is smaller than the maximum priority in the policy bank but is not same as any policy's priority

- . gotoPriorityExpression evaluates to a priority that is smaller than the current policy's priority

- If the gotoPriorityExpression evaluates to the priority of the current policy then the next policy in the priority order is evaluated.

- If the gotoPriorityExpression evaluates to the priority of a policy further ahead in the list then that policy will be evaluated next.

This field is applicable only to rewrite and responder policies.

type

The bindpoint to which the policy is bound

vsvrbindsvcip

used for showing the ip of bound entities

vsvrbindsvcport

used for showing ports of bound entities

gslbBoundSvcType

Protocol used by services bound to the GSLBvirtual server.

sitePersistCookie

This field is introduced for displaying the cookie in cluster setup.

svcSitePersistence

Type of Site Persistence set on the bound service

devno

count

Example

```
show gslb vserver gvip
```

stat gslb vserver

Displays statistics associated with a global server load balancing (GSLB) virtual server.

Synopsys

stat gslb vserver [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

name

Name of the GSLB virtual server for which to display statistics. If you do not specify a name, statistics are displayed for all GSLB virtual servers.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Current Client Est connections (CIntEstConn)

Number of client connections in ESTABLISHED state.

total INACTIVE services (inactSvcs)

number of INACTIVE services bound to a vserver

Vserver Health (Health)

Health of the vserver. This gives percentage of UP services bound to this vserver.

Vserver protocol (Protocol)

Protocol associated with the vserver

State

Current state of the server. Possible values are UP, DOWN, UNKNOWN, OFS(Out of Service), TROFS (Transition Out of Service), TROFS_DOWN(Down When going Out of Service)

total ACTIVE services (actSvcs)

number of ACTIVE services bound to a vserver

Vserver hits (Hits)

Total vserver hits

Request bytes (Reqb)

Total number of request bytes received on this service or virtual server.

Response bytes (Rspb)

Number of response bytes received by this service or virtual server.

Spill Over Threshold (SOTresh)

Spill Over Threshold set on the VServer.

Spill Over Hits (NumSo)

Number of times vserver experienced spill over.

Requests (Req)

Total number of requests received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

Responses (Rsp)

Number of responses received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

Current client connections (CIntConn)

Number of current client connections.

Current server connections (SvrConn)

Number of current connections to the actual servers behind the virtual server.

Current Server Est connections (SvrEstConn)

Number of server connections in ESTABLISHED state.

rename gslb vserver

Renames a global server load balancing (GSLB) virtual server.

Synopsis

```
rename gslb vserver <name>@ <newName>@
```

Arguments

name

Existing name of the GSLB virtual server.

newName

New name for the GSLB virtual server.

Example

```
rename gslb vserver gsl_vsvr gslb_vsvr_new
```


High Availability Commands

The entities on which you can perform NetScaler CLI operations:

- [HA failover](#)
- [HA files](#)
- [HA node](#)
- [HA sync](#)

HA failover

The following operations can be performed on "HA failover":

force HA failover

Forces an HA failover. Can be initiated from either node. A forced failover is not propagated or synchronized., Note: This command fails under any of the following conditions: * The secondary node is disabled or configured to remain secondary. * The primary node is configured to remain primary. * The state of the peer node is unknown. * You run the command on a standalone appliance.

Synopsys

force HA failover [-force]

Arguments

force

Force a failover without prompting for confirmation.

HA files

The following operations can be performed on "HA files":

sync HA files

Synchronize various configuration files from the primary node to the secondary. You can run this command from either node. Files that are present on only the secondary and are specific to the secondary are not deleted. This command fails if the secondary node is disabled, the secondary node is not accessible from the primary, or you enter the command on a standalone appliance.

Synopsys

sync HA files [<Mode> ...]

Arguments

Mode

Specify one of the following modes of synchronization.

- * all - Synchronize files related to system configuration, Access Gateway bookmarks, SSL certificates, SSL CRL lists, HTML injection scripts, and Application Firewall XML objects.
- * bookmarks - Synchronize all Access Gateway bookmarks.
- * ssl - Synchronize all certificates, keys, and CRLs for the SSL feature.
- * htmlinjection. Synchronize all scripts configured for the HTML injection feature.
- * imports. Synchronize all XML objects (for example, WSDLs, schemas, error pages) configured for the application firewall.
- * misc - Synchronize all license files and the rc.conf file.
- * all_plus_misc - Synchronize files related to system configuration, Access Gateway bookmarks, SSL certificates, SSL CRL lists, HTML injection scripts, application firewall XML objects, licenses, and the rc.conf file.

Example

```
sync files all
```

HA node

The following operations can be performed on "HA node":

add | **rm** | **set** | **unset** | **bind** | **unbind** | **show** | **stat**

add HA node

Adds a peer node to an HA configuration. Each node must add the other as a peer. An algorithm determines which node becomes primary and which becomes secondary.

Synopsys

add HA node <id> <IPAddress> [-inc (ENABLED | DISABLED)]

Arguments

id

Number that uniquely identifies the node. For self node, it will always be 0. Peer node values can range from 1-64.

Minimum value: 1

Maximum value: 64

IPAddress

The NSIP or NSIP6 address of the node to be added for an HA configuration. This setting is neither propagated nor synchronized.

inc

This option is required if the HA nodes reside on different networks. When this mode is enabled, the following independent network entities and configurations are neither propagated nor synced to the other node: MIPs, SNIPs, VLANs, routes (except LLB routes), route monitors, RNAT rules (except any RNAT rule with a VIP as the NAT IP), and dynamic routing configurations. They are maintained independently on each node.

Possible values: ENABLED, DISABLED

Default value: DISABLED

rm HA node

Removes the peer node from the HA configuration. To completely remove both the nodes from the HA configuration, you have to log on to each node and remove its peer node.

Synopsys

rm HA node <id>

Arguments

id

Number that uniquely identifies the peer node.

CLI users: To learn the ID of the peer node, run the show HA node command on the local node.

Minimum value: 0

Maximum value: 64

set HA node

Sets the specified HA related parameters for the node. The settings are neither propagated nor synchronized to the peer node.

Synopsys

```
set HA node [-haStatus <haStatus>] [-haSync ( ENABLED | DISABLED )] [-haProp ( ENABLED | DISABLED )] [-helloInterval <msecs>] [-deadInterval <secs>] [-failSafe ( ON | OFF )] [-maxFlips <positive_integer>] [-maxFlipTime <positive_integer>] [-syncvlan <positive_integer>]
```

Arguments

haStatus

The HA status of the node. The HA status STAYSECONDARY is used to force the secondary device stay as secondary independent of the state of the Primary device. For example, in an existing HA setup, the Primary node has to be upgraded and this process would take few seconds. During the upgradation, it is possible that the Primary node may suffer from a downtime for a few seconds. However, the Secondary should not take over as the Primary node. Thus, the Secondary node should remain as Secondary even if there is a failure in the Primary node.

STAYPRIMARY configuration keeps the node in primary state in case if it is healthy, even if the peer node was the primary node initially. If the node with STAYPRIMARY setting (and no peer node) is added to a primary node (which has this node as the peer) then this node takes over as the new primary and the older node becomes secondary. ENABLED state means normal HA operation without any constraints/preferences. DISABLED state disables the normal HA operation of the node.

Possible values: ENABLED, STAYSECONDARY, DISABLED, STAYPRIMARY

haSync

Automatically maintain synchronization by duplicating the configuration of the primary node on the secondary node. This setting is not propagated. Automatic synchronization requires that this setting be enabled (the default) on the current secondary node. Synchronization uses TCP port 3010.

Possible values: ENABLED, DISABLED

Default value: ENABLED

haProp

Automatically propagate all commands from the primary to the secondary node, except the following:

- * All HA configuration related commands. For example, add ha node, set ha node, and bind ha node.
- * All Interface related commands. For example, set interface and unset interface.
- * All channels related commands. For example, add channel, set channel, and bind channel.

The propagated command is executed on the secondary node before it is executed on the primary. If command propagation fails, or if command execution fails on the secondary, the primary node executes the command and logs an error. Command propagation uses port 3010.

Note: After enabling propagation, run force synchronization on either node.

Possible values: ENABLED, DISABLED

Default value: ENABLED

helloInterval

Interval, in milliseconds, between heartbeat messages sent to the peer node. The heartbeat messages are UDP packets sent to port 3003 of the peer node.

Default value: 200

Minimum value: 200

Maximum value: 1000

deadInterval

Number of seconds after which a peer node is marked DOWN if heartbeat messages are not received from the peer node.

Default value: 3

Minimum value: 3

Maximum value: 60

failSafe

Keep one node primary if both nodes fail the health check, so that a partially available node can back up data and handle traffic. This mode is set independently on each node.

Possible values: ON, OFF

Default value: OFF

maxFlips

Max number of flips allowed before becoming sticky primary

Default value: 0

Minimum value: 0

maxFlipTime

Interval after which flipping of node states can again start

Default value: 0

Minimum value: 0

syncvlan

Vlan on which HA related communication is sent. This include sync, propagation , connection mirroring , LB persistency config sync, persistent session sync and session state sync. However HA heartbeats can go all interfaces.

Minimum value: 1

Maximum value: 4094

unset HA node

Use this command to remove HA node settings. Refer to the set HA node command for meanings of the arguments.

Synopsys

```
unset HA node [-haStatus] [-haSync] [-haProp] [-helloInterval] [-deadInterval] [-failSafe] [-maxFlips] [-maxFlipTime] [-syncvlan]
```

bind HA node

Adds a route monitor to the local node. When a NetScaler appliance has only static routes for reaching a network, and you want to create a route monitor for the network, you must enable monitored static routes (MSR) for the static routes. Route Monitors are supported both in non-INC and INC modes.

Synopsys

```
bind HA node [<id>] (-routeMonitor <ip_addr|ipv6_addr|*> [<netmask>])
```

Arguments

id

Number that uniquely identifies the local node. The ID of the local node is always 0.

Minimum value: 0

Maximum value: 64

routeMonitor

A route that you want the NetScaler appliance to monitor in its internal routing table. You can specify an IPv4 address or network, or an IPv6 address or network prefix. If you specify an IPv4 network address or IPv6 network prefix, the appliance monitors any route that matches the network or prefix.

netmask

Subnet mask associated with the IPv4 route specified by the routeMonitor parameter.

unbind HA node

Removes a route monitor entry from the local node. The NetScaler appliance stops monitoring the route in its internal routing table.

Synopsys

unbind HA node [<id>] (-routeMonitor <ip_addr|ipv6_addr*> [<netmask>])

Arguments

id

Number that uniquely identifies the local node. The ID of the local node is always 0.

Minimum value: 0

Maximum value: 64

routeMonitor

The route specified in the route monitor entry that you want to remove from the NetScaler appliance. Can be an IPv4 address or network, or an IPv6 address or network prefix.

netmask

Subnet mask associated with the IPv4 route specified by the routeMonitor parameter.

show HA node

Displays the HA settings of both nodes or, if you specify a node, just the specified node. You can use this command to display the master state (primary or secondary) of the nodes in a HA configuration.

Synopsys

show HA node [<id>]

Arguments

id

ID of the node whose HA settings you want to display. (The ID of the local node is always 0.)

Minimum value: 0

Maximum value: 64

Outputs

name

Node Name.

IPAddress

IP Address of the node.

flags

The flags for this entry.

stateflag

haStatus

HA status.

state

HA Master State.

haSync

HA Sync State.

haProp

HA Propagation Status.

enaifaces

Enabled interfaces.

disifaces

Disabled interfaces.

hamonifaces

HAMON ON interfaces.

pfifaces

Interfaces causing Partial Failure.

ifaces

Interfaces on which non-multicast is not seen.

network

The network.

netmask

The netmask.

inc

INC state.

ssl2

SSL card status.

helloInterval

Hello Interval.

deadInterval

Dead Interval.

masterStateTime

Time elapsed in current master state

failSafe

Keep one node primary if both nodes fail the health check, so that a partially available node can back up data and handle traffic. This mode is set independently on each node.

routeMonitor

The IP address (IPv4 or IPv6).

maxFlips

Max number of flips allowed before becoming sticky primary

maxFlipTime

Interval after which flipping of node states can again start

curFlips

Keeps track of number of flips that have happened till now in current interval

completedFlipTime

To inform user whether flip time is elapsed or not

syncvlan

Vlan on which HA related communication is sent. This include sync, propagation , connection mirroring , LB persistency config sync, persistent session sync and session state sync. However HA heartbeats can go all interfaces.

routeMonitorState

State for route monitor

devno**count**

Example

An example of the command's output is as follows: 2 configured nodes: 1) Node ID: 0 IP: 1!

stat HA node

Display the statistics related to HA configuration.

Synopsys

stat HA node [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

High Availability (HA)

Whether a NetScaler appliance is configured for high availability. Possible values are YES and NO. If the value is NO, the high availability statistics below are invalid.

System state (HAState)

State of the HA node, based on its health, in a high availability setup. Possible values are:

UP ? Indicates that the node is accessible and can function as either a primary or secondary node.

DISABLED ? Indicates that the high availability status of the node has been manually disabled. Synchronization and propagation cannot take place between the peer nodes.

INIT ? Indicates that the node is in the process of becoming part of the high availability configuration.

PARTIALFAIL ? Indicates that one of the high availability monitored interfaces has failed because of a card or link failure. This state triggers a failover.

COMPLETEFAIL ? Indicates that all the interfaces of the node are unusable, because the interfaces on which high availability monitoring is enabled are not connected or are manually disabled. This state triggers a failover.

DUMB ? Indicates that the node is in listening mode. It does not participate in high availability transitions or transfer configuration from the peer node. This is a configured value, not a statistic.

PARTIALFAILSSL ? Indicates that the SSL card has failed. This state triggers a failover.

ROUTEMONITORFAIL ? Indicates that the route monitor has failed. This state triggers a failover.

Master state (mastate)

Indicates the high availability state of the node. Possible values are:

STAYSECONDARY ? Indicates that the selected node remains the secondary node in a high availability setup. In this case a forced failover does not change the state but, instead, returns an appropriate error message. This is a configured value and not a statistic.

PRIMARY ? Indicates that the selected node is the primary node in a high availability setup.

SECONDARY ? Indicates that the selected node is the secondary node in a high availability setup.

CLAIMING ? Indicates that the secondary node is in the process of taking over as the primary node. This is the intermediate state in the transition of the secondary node to primary status.

FORCE CHANGE - Indicates that the secondary node is forcibly changing its status to primary due to a forced failover issued on the secondary node.

Last Transition time (TransTime)

Time when the last master state transition occurred. You can use this statistic for debugging.

Heartbeats received (HApktrx)

Number of heartbeat packets received from the peer node. Heartbeats are sent at regular intervals (default is 200 milliseconds) to determine the state of the peer node.

Heartbeats sent (HApkttx)

Number of heartbeat packets sent to the peer node. Heartbeats are sent at regular intervals (default is 200 milliseconds) to determine the state of the peer node.

Propagation timeouts (ptimeout)

Number of times propagation timed out.

Sync failure (syncfail)

Number of times the configuration of the primary and secondary nodes failed to synchronize since that last transition. A synchronization failure results in mismatched configuration. It can be caused by a mismatch in the Remote Procedural Call (RPC) password on the two nodes forming the high availability pair.

HA sync

The following operations can be performed on "HA sync":

force HA sync

Forces duplication of the primary node's configuration on the secondary node. Can be executed from either node. Note: This command fails under any of the following conditions: * Synchronization is already in progress. * The secondary node is disabled. * Synchronization is disabled on either node * The secondary node is not accessible from the primary. * You run the command on a standalone appliance.

Synopsys

```
force HA sync [-force [-save ( YES | NO )]]
```

Arguments

force

Force synchronization regardless of the state of HA propagation and HA synchronization on either node.

save

After synchronization, automatically save the configuration in the secondary node configuration file (ns.conf) without prompting for confirmation.

Possible values: YES, NO

Default value: VAL_NOT_SET

Example

Can be used in following formats: >force sync <cr> >force sync -force <cr>

IPSec Commands

The entities on which you can perform NetScaler CLI operations:

- [ipsec counters](#)
- [ipsec parameter](#)
- [ipsec profile](#)

ipsec counters

The following operations can be performed on "ipsec counters":

stat ipsec counters

Display statistics for secure tunnel sessions.

Synopsys

```
stat ipsec counters [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

Bytes Received (ipsecRxBytes)

Bytes received during IPsec sessions.

Bytes Sent (ipsecTxBytes)

Bytes sent during IPsec sessions.

Packets Received (ipsecRxPkts)

Packets received during IPsec sessions.

Packets Sent (ipsecTxPkts)

Packets sent during IPsec sessions.

Example

```
stat ipsec
```

ipsec parameter

The following operations can be performed on "ipsec parameter":

[set](#) | [unset](#) | [show](#)

set ipsec parameter

Set global parameters for IPSEC

Synopsys

```
set ipsec parameter [-ikeVersion ( V1 | V2 )] [-encAlgo ( AES | 3DES ) ...] [-hashAlgo <hashAlgo> ...] [-lifetime <positive_integer>] [-livenessCheckInterval <positive_integer>] [-replayWindowSize <positive_integer>] [-ikeRetryInterval <positive_integer>] [-perfectForwardSecrecy ( ENABLE | DISABLE )] [-retransmissiontime <positive_integer>]
```

Arguments

ikeVersion

IKE Protocol Version

Possible values: V1, V2

Default value: V2

encAlgo

Type of encryption algorithm

Default value: AES

hashAlgo

Type of hashing algorithm

Default value: HMAC_SHA256

lifetime

Lifetime of IKE SA in seconds. Lifetime of IPSec SA will be (lifetime of IKE SA/8)

Minimum value: 480

Maximum value: 31536000

livenessCheckInterval

Number of seconds after which a notify payload is sent to check the liveness of the peer. Additional retries are done as per retransmit interval setting. Zero value disables liveness checks.

Minimum value: 0

Maximum value: 64999

replayWindowSize

IPSec Replay window size for the data traffic

Minimum value: 0

Maximum value: 16384

ikeRetryInterval

IKE retry interval for bringing up the connection

Minimum value: 60

Maximum value: 3600

perfectForwardSecrecy

Enable/Disable PFS.

Possible values: ENABLE, DISABLE

Default value: DISABLE

retransmissiontime

The interval in seconds to retry sending the IKE messages to peer, three consecutive attempts are done with doubled interval after every failure.,

increases for every retransmit till 6 retransmits.

Minimum value: 1

Maximum value: 99

unset ipsec parameter

Set global parameters for IPSEC. Refer to the set ipsec parameter command for meanings of the arguments.

Synopsys

unset ipsec parameter [-ikeVersion] [-encAlgo] [-hashAlgo] [-lifetime] [-livenessCheckInterval] [-replayWindowSize] [-ikeRetryInterval] [-perfectForwardSecrecy] [-retransmissiontime]

show ipsec parameter

Show global parameters for IPSEC

Synopsys

show ipsec parameter

Outputs

ikeVersion

IKE Protocol Version

encAlgo

Type of encryption algorithm

hashAlgo

Type of hashing algorithm

lifetime

Lifetime of IKE SA in seconds. Lifetime of IPSec SA will be (lifetime of IKE SA/8)

livenessCheckInterval

Number of seconds after which a notify payload is sent to check the liveness of the peer. Additional retries are done as per retransmit interval setting. Zero value disables liveness checks.

replayWindowSize

IPSec Replay window size for the data traffic

ikeRetryInterval

IKE retry interval for bringing up the connection

perfectForwardSecrecy

Enable/Disable PFS.

retransmissiontime

The interval in seconds to retry sending the IKE messages to peer, three consecutive attempts are done with doubled interval after every failure.,

increases for every retransmit till 6 retransmits.

ipsec profile

The following operations can be performed on "ipsec profile":

add | **show** | **rm**

add ipsec profile

Add an ipsec profile.

Synopsys

```
add ipsec profile <name> [-ikeVersion ( V1 | V2 )] [-encAlgo ( AES | 3DES ) ...] [-hashAlgo <hashAlgo> ...] [-lifetime
<positive_integer>] (-psk | (-publickey <string> -privatekey <string> -peerPublicKey <string>)) [-
livenessCheckInterval <positive_integer>] [-replayWindowSize <positive_integer>] [-ikeRetryInterval
<positive_integer>] [-retransmissiontime <positive_integer>] [-perfectForwardSecrecy ( ENABLE | DISABLE )]
```

Arguments

name

The name of the ipsec profile

ikeVersion

IKE Protocol Version

Possible values: V1, V2

encAlgo

Type of encryption algorithm

hashAlgo

Type of hashing algorithm

lifetime

Lifetime of IKE SA in seconds. Lifetime of IPSec SA will be (lifetime of IKE SA/8)

Minimum value: 480

Maximum value: 31536000

psk

Pre shared key value

publickey

Public key file path

privatekey

Private key file path

peerPublicKey

Peer public key file path

livenessCheckInterval

Number of seconds after which a notify payload is sent to check the liveness of the peer. Additional retries are done as per retransmit interval setting. Zero value disables liveness checks.

Minimum value: 0

Maximum value: 64999

replayWindowSize

IPSec Replay window size for the data traffic

Minimum value: 0

Maximum value: 16384

ikeRetryInterval

IKE retry interval for bringing up the connection

Minimum value: 60

Maximum value: 3600

retransmissiontime

The interval in seconds to retry sending the IKE messages to peer, three consecutive attempts are done with doubled interval after every failure.

Minimum value: 1

Maximum value: 99

perfectForwardSecrecy

Enable/Disable PFS.

Possible values: ENABLE, DISABLE

show ipsec profile

Display all of the configured ipsec peers

Synopsys

show ipsec profile [<name>]

Arguments

name

The name of the ipsec profile

Outputs

ikeVersion

IKE Protocol Version

encAlgo

Type of encryption algorithm.

hashAlgo

Type of hashing algorithm

lifetime

Lifetime of IKE SA in seconds. Lifetime of IPSec SA will be (lifetime of IKE SA/8)

livenessCheckInterval

Number of seconds after which a notify payload is sent to check the liveness of the peer. Additional retries are done as per retransmit interval setting. Zero value disables liveness checks.

replayWindowSize

IPSec Replay window size for the data traffic

retransmissiontime

The interval in seconds to retry sending the IKE messages to peer, three consecutive attempts are done with doubled interval after every failure.

psk

Pre shared key value

publickey

Public key file path

privatekey

Private key file path

peerPublicKey

Peer public key file path

ikeRetryInterval

IKE retry interval for bringing up the connection

perfectForwardSecrecy

Enable/Disable PFS.

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

devno**count****stateflag**

Example

```
show ipsec profile
```

rm ipsec profile

Remove an ipsec peer

Synopsys

```
rm ipsec profile <name>
```

Arguments

name

The name of the ipsec profile.

Example

```
rm ipsec profile
```

Load Balancing Commands

The entities on which you can perform NetScaler CLI operations:

- o lb group
- o lb metricTable
- o lb monbindings
- o lb monitor
- o lb parameter
- o lb persistentSessions
- o lb route
- o lb route6
- o lb sipParameters
- o lb vserver
- o lb wlm

lb group

The following operations can be performed on "lb group":

set | **unset** | **bind** | **unbind** | **show** | **rename**

set lb group

Configures persistence for the specified load balancing group. The persistence settings are applied to all the members of the group.

Synopsis

```
set lb group <name>@ [-persistenceType <persistenceType>] [-persistenceBackup ( SOURCEIP | NONE )] [-  
backupPersistenceTimeout <mins>] [-persistMask <netmask>] [-cookieName <string>] [-v6persistmasklen  
<positive_integer>] [-cookieDomain <string>] [-timeout <mins>] [-rule <expression>]
```

Arguments

name

Name of the load balancing virtual server group.

persistenceType

Type of persistence for the group. Available settings function as follows:

- * SOURCEIP - Create persistence sessions based on the client IP.
- * COOKIEINSERT - Create persistence sessions based on a cookie in client requests. The cookie is inserted by a Set-Cookie directive from the server, in its first response to a client.
- * RULE - Create persistence sessions based on a user defined rule.
- * NONE - Disable persistence for the group.

Possible values: SOURCEIP, COOKIEINSERT, RULE, NONE

persistenceBackup

Type of backup persistence for the group.

Possible values: SOURCEIP, NONE

backupPersistenceTimeout

Time period, in minutes, for which backup persistence is in effect.

Default value: 2

Minimum value: 2

Maximum value: 1440

persistMask

Persistence mask to apply to source IPv4 addresses when creating source IP based persistence sessions.

Default value: 0xFFFFFFFF

cookieName

Use this parameter to specify the cookie name for COOKIE persistence type. It specifies the name of cookie with a maximum of 32 characters. If not specified, cookie name is internally generated.

v6persistmasklen

Persistence mask to apply to source IPv6 addresses when creating source IP based persistence sessions.

Default value: 128

Minimum value: 1

Maximum value: 128

cookieDomain

Domain attribute for the HTTP cookie.

timeout

Time period for which a persistence session is in effect.

Default value: 2

Maximum value: 1440

rule

Expression, or name of a named expression, against which traffic is evaluated. Written in the classic or default syntax.

Note:

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

The following requirements apply only to the NetScaler CLI:

- * If the expression includes one or more spaces, enclose the entire expression in double quotation marks.
- * If the expression itself includes double quotation marks, escape the quotations by using the \ character.
- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

Default value: "None"

Example

```
set lb group webgrp -persistenceType COOKIEINSERT To set the persistence type for multip.
```

unset lb group

Use this command to remove lb group settings. Refer to the set lb group command for meanings of the arguments.

Synopsys

```
unset lb group <name>@ [-persistenceType] [-persistenceBackup] [-backupPersistenceTimeout] [-persistMask] [-cookieName] [-v6persistmasklen] [-cookieDomain] [-timeout] [-rule]
```

bind lb group

Binds one or more virtual servers to a load balancing virtual server group. If the specified group does not exist, the NetScaler appliance first creates the group, and then binds the virtual servers to it. A virtual server group enables you to specify common persistence settings for all of its members through a single set lb group command. Only address-based virtual servers can be added to a group. Content-based virtual servers (content switching and cache redirection virtual servers) cannot be added. A virtual server can be assigned to only one group at any given time. To move a virtual server from one group to another, the virtual server must first be unbound from the group to which it belongs.

Synopsys

```
bind lb group <name>@ <vServerName>@ ...
```

Arguments

name

Name for the load balancing virtual server group. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Can be changed after the virtual server is created.

CLI Users: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my lbgroup" or 'my lbgroup').

vServerName

Name of the virtual server to bind to the group. Multiple names can be specified.

Example

```
bind lb group webgrp http_vip To bind multiple vservers to a group use the following com
```

unbind lb group

Unbinds one or more virtual servers from a group. When the last virtual server is unbound, the group is removed.

Synopsys

```
unbind lb group <name> <vServerName>@ ...
```

Arguments

name

Name of the load balancing virtual server group.

vServerName

Name of the virtual server to unbind. Multiple names can be specified.

Example

```
unbind lb group webgroup http_vip To unbind multiple vservers use the following command:
```

show lb group

Displays the virtual servers bound to the specified group.

Synopsys

```
show lb group [<name>]
```

Arguments

name

Name of the load balancing virtual server group.

Outputs

vServerName

Virtual server name.

persistenceType

The type of the persistence set for the group.

persistenceBackup

The type of the backup persistence set for the group.

backupPersistenceTimeout

Time period, in minutes, for which backup persistence is in effect.

persistMask

The netmask applied for ipv4 traffic when the persistency type is SOURCEIP.

v6persistmasklen

The netmask applied for ipv6 traffic when the persistency type is SOURCEIP.

cookieName

Use this parameter to specify the cookie name for COOKIE persistence type. It specifies the name of cookie with a maximum of 32 characters. If not specified, cookie name is internally generated.

cookieDomain

Domain attribute for the HTTP cookie.

timeout

Time period for which a persistence session is in effect.

rule

Rule type.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

stateflag**devno****count**

Example

```
show lb group webgrp
```

rename lb group

Renames a load balancing virtual server group.

Synopsys

```
rename lb group <name>@ <newName>@
```

Arguments

name

Existing name of the load balancing virtual server group.

newName

New name for the load balancing virtual server group.

Example

```
rename lb group gvl gv-new1
```

lb metricTable

The following operations can be performed on "lb metricTable":

add | **rm** | **set** | **bind** | **unbind** | **show**

add lb metricTable

Creates a metric table for load monitoring.

Synopsys

```
add lb metricTable <metricTable>
```

Arguments

metricTable

Name for the metric table. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

CLI Users: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my metrictable" or 'my metrictable').

Example

```
add metrictable newtable
```

rm lb metricTable

Removes a metric table.

Synopsys

```
rm lb metricTable <metricTable>
```

Arguments

metricTable

Name of the metric table.

Example

```
rm metric table netscaler
```

set lb metricTable

Modifies the SNMP OID of a metric in a metric table.

Synopsys

```
set lb metricTable <metricTable> <metric> <snmpOID>
```

Arguments

metricTable

Name of the metric table.

metric

Name of the metric for which to change the SNMP OID.

snmpOID

New SNMP OID of the metric.

Example

```
set metricTable table met1 aliasname oidstr
```

bind lb metricTable

Binds a metric to a metric table. You must also specify the SNMP OID of the metric.

Synopsys

```
bind lb metricTable <metricTable> <metric> <snmpOID>
```

Arguments

metricTable

Name of the metric table.

metric

Name of the metric.

snmpOID

SNMP OID of the metric.

Example

```
bind metricTable tablename aliasname 1.2.3.4
```

unbind lb metricTable

Unbinds a metric from a metric table.

Synopsys

```
unbind lb metricTable <metricTable> <metric>
```

Arguments

metricTable

Name of the metric table.

metric

Name of the metric to unbind.

Example

```
unbind metricTable tablename aliasname
```

show lb metricTable

Displays the parameters of the specified metric table. If no metric table name is specified, a list of all configured metric tables is displayed.

Synopsys

show lb metricTable [<metricTable>]

Arguments

metricTable

Name of the metric table.

Outputs

metric

Metric name of the oid.

snmpOID

OID corresponding to the metric

flags

flags controlling display

stateflag

flags controlling display

metricType

Indication if it is a configured or internal

type

Adds a temporary or permanent table.

devno

count

Example

An example of the show metrictable command output is as follows:

Name : ALTEON

Ib monbindings

The following operations can be performed on "Ib monbindings":

show Ib monbindings

Display the services to which this monitor is bound

Synopsys

show Ib monbindings <monitorName>

Arguments

monitorName

The name of the monitor.

Outputs

type

The type of monitor.

state

The state of the monitor.

boundServiceGroupSvrState

The state of the servicegroup.

monsvcState

The configured state (enable/disable) of Monitor on this service.

monState

The configured state (enable/disable) of Monitor on this service.

IPAddress

The IPAddress of the service.

port

The port of the service.

serviceName

The name of the service.

serviceName

The name of the service group.

serviceType

The type of service

svrState

The state of the service

stateflag

devno

count

lb monitor

The following operations can be performed on "lb monitor":

add | **rm** | **set** | **unset** | **enable** | **disable** | **bind** | **unbind** | **show**

add lb monitor

Creates a monitor that you can bind to load balancing services. The monitor periodically sends probes to those services to test their availability.

Synopsys

```
add lb monitor <monitorName> <type> [-action <action>] [-respCode <int[-int]> ...] [-httpRequest <string>] [-rtspRequest <string>] [-customHeaders <string>] [-maxForwards <positive_integer>] [-sipMethod <sipMethod>] [-sipURI <string>] [-sipregURI <string>] [-send <string>] [-recv <string>] [-query <string>] [-queryType <queryType>] [-scriptName <string>] [-scriptArgs <string>] [-dispatcherIP <ip_addr>] [-dispatcherPort <port>] [-userName <string>] [-password { -secondaryPassword }] [-logonpointName <string>] [-lasVersion <string>] [-radKey { -radNASid <string> }] [-radNASip <ip_addr>] [-radAccountType <positive_integer>] [-radFramedIP <ip_addr>] [-radAPN <string>] [-radMSISDN <string>] [-radAccountSession <string>] [-LRTM ( ENABLED | DISABLED )] [-deviation <positive_integer>] [-units <units>] [-interval <integer>] [-resptimeout <integer>] [-resptimeoutThresh <positive_integer>] [-retries <integer>] [-failureRetries <integer>] [-alertRetries <integer>] [-successRetries <integer>] [-downTime <integer>] [-destIP <ip_addr|ipv6_addr>] [-destPort <port>] [-state ( ENABLED | DISABLED )] [-reverse ( YES | NO )] [-transparent ( YES | NO )] [-ipTunnel ( YES | NO )] [-tos ( YES | NO )] [-tosId <positive_integer>] [-secure ( YES | NO )] [-validateCred ( YES | NO )] [-domain <string>] [-IPAddress <ip_addr|ipv6_addr|*> ...] [-group <string>] [-fileName <string>] [-baseDN <string>] [-bindDN <string>] [-filter <string>] [-attribute <string>] [-database <string>] [-oracleSid <string>] [-sqlQuery <text>] [-evalRule <expression>] [-mssqlProtocolVersion <mssqlProtocolVersion>] [-snmpOID <string>] [-snmpCommunity <string>] [-snmpThreshold <string>] [-snmpVersion ( V1 | V2 )] [-metricTable <string>] [-application <string>] [-sitePath <string>] [-storename <string>] [-storefrontacctservice ( YES | NO )] [-netProfile <string>] [-originHost <string>] [-originRealm <string>] [-hostIPAddress <ip_addr|ipv6_addr|*>] [-vendorId <positive_integer>] [-productName <string>] [-firmwareRevision <positive_integer>] [-authApplicationId <positive_integer> ...] [-acctApplicationId <positive_integer> ...] [-inbandSecurityId ( NO_INBAND_SECURITY | TLS )] [-supportedVendorIds <positive_integer> ...] [-vendorSpecificVendorId <positive_integer>] [-vendorSpecificAuthApplicationIds <positive_integer> ...] [-vendorSpecificAcctApplicationIds <positive_integer> ...] [-kcdAccount <string>] [-storedb ( ENABLED | DISABLED )] [-storefrontcheckbackendservices ( YES | NO )]
```

Arguments

monitorName

Name for the monitor. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

CLI Users: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my monitor" or 'my monitor').

type

Type of monitor that you want to create.

Possible values: PING, TCP, HTTP, TCP-ECV, HTTP-ECV, UDP-ECV, DNS, FTP, LDNS-PING, LDNS-TCP, LDNS-DNS, RADIUS, USER, HTTP-INLINE, SIP-UDP, LOAD, FTP-EXTENDED, SMTP, SNMP, NNTP, MYSQL, MYSQL-ECV, MSSQL-ECV, ORACLE-ECV, LDAP, POP3, CITRIX-XML-SERVICE, CITRIX-WEB-INTERFACE, DNS-TCP, RTSP, ARP, CITRIX-AG, CITRIX-AAC-LOGINPAGE, CITRIX-AAC-LAS, CITRIX-XD-DDC, ND6, CITRIX-WI-EXTENDED, DIAMETER, RADIUS_ACCOUNTING, STOREFRONT, APPC, CITRIX-XNC-ECV, CITRIX-XDM

action

Action to perform when the response to an inline monitor (a monitor of type HTTP-INLINE) indicates that the service is down. A service monitored by an inline monitor is considered DOWN if the response code is not one of the codes that have been specified for the Response Code parameter.

Available settings function as follows:

* NONE - Do not take any action. However, the show service command and the show lb monitor command indicate the total number of responses that were checked and the number of consecutive error responses received after the last successful probe.

* LOG - Log the event in NSLOG or SYSLOG.

* DOWN - Mark the service as being down, and then do not direct any traffic to the service until the configured down time has expired. Persistent connections to the service are terminated as soon as the service is marked as DOWN. Also, log the event in NSLOG or SYSLOG.

Possible values: NONE, LOG, DOWN

Default value: DOWN

respCode

Response codes for which to mark the service as UP. For any other response code, the action performed depends on the monitor type. HTTP monitors and RADIUS monitors mark the service as DOWN, while HTTP-INLINE monitors perform the action indicated by the Action parameter.

httpRequest

HTTP request to send to the server (for example, "HEAD /file.html").

rtspRequest

RTSP request to send to the server (for example, "OPTIONS *").

customHeaders

Custom header string to include in the monitoring probes.

maxForwards

Maximum number of hops that the SIP request used for monitoring can traverse to reach the server. Applicable only to monitors of type SIP-UDP.

Default value: 1

Minimum value: 0

Maximum value: 255

sipMethod

SIP method to use for the query. Applicable only to monitors of type SIP-UDP.

Possible values: OPTIONS, INVITE, REGISTER

sipURI

SIP URI string to send to the service (for example, sip:sip.test). Applicable only to monitors of type SIP-UDP.

sipregURI

SIP user to be registered. Applicable only if the monitor is of type SIP-UDP and the SIP Method parameter is set to REGISTER.

send

String to send to the service. Applicable to TCP-ECV, HTTP-ECV, and UDP-ECV monitors.

recv

String expected from the server for the service to be marked as UP. Applicable to TCP-ECV, HTTP-ECV, and UDP-ECV monitors.

query

Domain name to resolve as part of monitoring the DNS service (for example, example.com).

queryType

Type of DNS record for which to send monitoring queries. Set to Address for querying A records, AAAA for querying AAAA records, and Zone for querying the SOA record.

Possible values: Address, Zone, AAAA

scriptName

Path and name of the script to execute. The script must be available on the NetScaler appliance, in the /nsconfig/monitors/ directory.

scriptArgs

String of arguments for the script. The string is copied verbatim into the request.

dispatcherIP

IP address of the dispatcher to which to send the probe.

dispatcherPort

Port number on which the dispatcher listens for the monitoring probe.

userName

User name with which to probe the RADIUS, NNTP, FTP, FTP-EXTENDED, MYSQL, MSSQL, POP3, CITRIX-AG, CITRIX-XD-DDC, CITRIX-WI-EXTENDED, CITRIX-XNC or CITRIX-XDM server.

password

Password that is required for logging on to the RADIUS, NNTP, FTP, FTP-EXTENDED, MYSQL, MSSQL, POP3, CITRIX-AG, CITRIX-XD-DDC, CITRIX-WI-EXTENDED, CITRIX-XNC-ECV or CITRIX-XDM server. Used in conjunction with the user name specified for the User Name parameter.

secondaryPassword

Secondary password that users might have to provide to log on to the Access Gateway server. Applicable to CITRIX-AG monitors.

logonpointName

Name of the logon point that is configured for the Citrix Access Gateway Advanced Access Control software. Required if you want to monitor the associated login page or Logon Agent. Applicable to CITRIX-AAC-LAS and CITRIX-AAC-LOGINPAGE monitors.

lasVersion

Version number of the Citrix Advanced Access Control Logon Agent. Required by the CITRIX-AAC-LAS monitor.

radKey

Authentication key (shared secret text string) for RADIUS clients and servers to exchange. Applicable to monitors of type RADIUS and RADIUS_ACCOUNTING.

radNASid

NAS-Identifier to send in the Access-Request packet. Applicable to monitors of type RADIUS.

radNASip

Network Access Server (NAS) IP address to use as the source IP address when monitoring a RADIUS server. Applicable to monitors of type RADIUS and RADIUS_ACCOUNTING.

radAccountType

Account Type to be used in Account Request Packet. Applicable to monitors of type RADIUS_ACCOUNTING.

Default value: 1

Minimum value: 0

Maximum value: 15

radFramedIP

Source ip with which the packet will go out . Applicable to monitors of type RADIUS_ACCOUNTING.

radAPN

Called Station Id to be used in Account Request Packet. Applicable to monitors of type RADIUS_ACCOUNTING.

radMSISDN

Calling Stations Id to be used in Account Request Packet. Applicable to monitors of type RADIUS_ACCOUNTING.

radAccountSession

Account Session ID to be used in Account Request Packet. Applicable to monitors of type RADIUS_ACCOUNTING.

LRTM

Calculate the least response times for bound services. If this parameter is not enabled, the appliance does not learn the response times of the bound services. Also used for LRTM load balancing.

Possible values: ENABLED, DISABLED

deviation

Time value added to the learned average response time in dynamic response time monitoring (DRTM). When a deviation is specified, the appliance learns the average response time of bound services and adds the deviation to the average. The final value is then continually adjusted to accommodate response time variations over time. Specified in milliseconds, seconds, or minutes.

Minimum value: 0

Maximum value: 20939000

units

Unit of measurement for the Down Time parameter. Cannot be changed after the monitor is created.

Possible values: SEC, MSEC, MIN

Default value: SEC

interval

Time interval between two successive probes. Must be greater than the value of Response Time-out.

Default value: 5

Minimum value: 1

Maximum value: 20940000

resptimeout

Amount of time for which the appliance must wait before it marks a probe as FAILED. Must be less than the value specified for the Interval parameter.

Note: For UDP-ECV monitors for which a receive string is not configured, response timeout does not apply. For UDP-ECV monitors with no receive string, probe failure is indicated by an ICMP port unreachable error received from the service.

Default value: 2

Minimum value: 1

Maximum value: 20939000

resptimeoutThresh

Response time threshold, specified as a percentage of the Response Time-out parameter. If the response to a monitor probe has not arrived when the threshold is reached, the appliance generates an SNMP trap called `monRespTimeoutAboveThresh`. After the response time returns to a value below the threshold, the appliance generates a `monRespTimeoutBelowThresh` SNMP trap. For the traps to be generated, the "MONITOR-RTO-THRESHOLD" alarm must also be enabled.

Minimum value: 0

Maximum value: 100

retries

Maximum number of probes to send to establish the state of a service for which a monitoring probe failed.

Default value: 3

Minimum value: 1

Maximum value: 127

failureRetries

Number of retries that must fail, out of the number specified for the Retries parameter, for a service to be marked as DOWN. For example, if the Retries parameter is set to 10 and the Failure Retries parameter is set to 6, out of the ten probes sent, at least six probes must fail if the service is to be marked as DOWN. The default value of 0 means that all the retries must fail if the service is to be marked as DOWN.

Maximum value: 32

alertRetries

Number of consecutive probe failures after which the appliance generates an SNMP trap called `monProbeFailed`.

Maximum value: 32

successRetries

Number of consecutive successful probes required to transition a service's state from DOWN to UP.

Default value: 1

Minimum value: 1

Maximum value: 32

downTime

Time duration for which to wait before probing a service that has been marked as DOWN. Expressed in milliseconds, seconds, or minutes.

Default value: 30

Minimum value: 1

Maximum value: 20939000

destIP

IP address of the service to which to send probes. If the parameter is set to 0, the IP address of the server to which the monitor is bound is considered the destination IP address.

destPort

TCP or UDP port to which to send the probe. If the parameter is set to 0, the port number of the service to which the monitor is bound is considered the destination port. For a monitor of type USER, however, the destination port is the port number that is included in the HTTP request sent to the dispatcher. Does not apply to monitors of type PING.

state

State of the monitor. The DISABLED setting disables not only the monitor being configured, but all monitors of the same type, until the parameter is set to ENABLED. If the monitor is bound to a service, the state of the monitor is not taken into account when the state of the service is determined.

Possible values: ENABLED, DISABLED

Default value: ENABLED

reverse

Mark a service as DOWN, instead of UP, when probe criteria are satisfied, and as UP instead of DOWN when probe criteria are not satisfied.

Possible values: YES, NO

Default value: NO

transparent

The monitor is bound to a transparent device such as a firewall or router. The state of a transparent device depends on the responsiveness of the services behind it. If a transparent device is being monitored, a destination IP address must be specified. The probe is sent to the specified IP address by using the MAC address of the transparent device.

Possible values: YES, NO

Default value: NO

ipTunnel

Send the monitoring probe to the service through an IP tunnel. A destination IP address must be specified.

Possible values: YES, NO

Default value: NO

tos

Probe the service by encoding the destination IP address in the IP TOS (6) bits.

Possible values: YES, NO

tosId

The TOS ID of the specified destination IP. Applicable only when the TOS parameter is set.

Minimum value: 1

Maximum value: 63

secure

Use a secure SSL connection when monitoring a service. Applicable only to TCP based monitors. The secure option cannot be used with a CITRIX-AG monitor, because a CITRIX-AG monitor uses a secure connection by default.

Possible values: YES, NO

Default value: NO

validateCred

Validate the credentials of the Xen Desktop DDC server user. Applicable to monitors of type CITRIX-XD-DDC.

Possible values: YES, NO

Default value: NO

domain

Domain in which the XenDesktop Desktop Delivery Controller (DDC) servers or Web Interface servers are present. Required by CITRIX-XD-DDC and CITRIX-WI-EXTENDED monitors for logging on to the DDC servers and Web Interface servers, respectively.

IPAddress

Set of IP addresses expected in the monitoring response from the DNS server, if the record type is A or AAAA. Applicable to DNS monitors.

group

Name of a newsgroup available on the NNTP service that is to be monitored. The appliance periodically generates an NNTP query for the name of the newsgroup and evaluates the response. If the newsgroup is found on the server, the service is marked as UP. If the newsgroup does not exist or if the search fails, the service is marked as DOWN. Applicable to NNTP monitors.

fileName

Name of a file on the FTP server. The appliance monitors the FTP service by periodically checking the existence of the file on the server. Applicable to FTP-EXTENDED monitors.

baseDN

The base distinguished name of the LDAP service, from where the LDAP server can begin the search for the attributes in the monitoring query. Required for LDAP service monitoring.

bindDN

The distinguished name with which an LDAP monitor can perform the Bind operation on the LDAP server. Optional. Applicable to LDAP monitors.

filter

Filter criteria for the LDAP query. Optional.

attribute

Attribute to evaluate when the LDAP server responds to the query. Success or failure of the monitoring probe depends on whether the attribute exists in the response. Optional.

database

Name of the database to connect to during authentication.

oracleSid

Name of the service identifier that is used to connect to the Oracle database during authentication.

sqlQuery

SQL query for a MYSQL-ECV or MSSQL-ECV monitor. Sent to the database server after the server authenticates the connection.

evalRule

Default syntax expression that evaluates the database server's response to a MYSQL-ECV or MSSQL-ECV monitoring query. Must produce a Boolean result. The result determines the state of the server. If the expression returns TRUE, the probe succeeds.

For example, if you want the appliance to evaluate the error message to determine the state of the server, use the rule `MYSQL.RES.ROW(10).TEXT_ELEM(2).EQ("MySQL")`.

mssqlProtocolVersion

Version of MSSQL server that is to be monitored.

Possible values: 70, 2000, 2000SP1, 2005, 2008, 2008R2, 2012

Default value: 70

snmpOID

SNMP OID for SNMP monitors.

snmpCommunity

Community name for SNMP monitors.

snmpThreshold

Threshold for SNMP monitors.

snmpVersion

SNMP version to be used for SNMP monitors.

Possible values: V1, V2

metricTable

Metric table to which to bind metrics.

application

Name of the application used to determine the state of the service. Applicable to monitors of type CITRIX-XML-SERVICE.

sitePath

URL of the logon page. For monitors of type CITRIX-WEB-INTERFACE, to monitor a dynamic page under the site path, terminate the site path with a slash (/). Applicable to CITRIX-WEB-INTERFACE, CITRIX-WI-EXTENDED and CITRIX-XDM monitors.

storename

Store Name. For monitors of type STOREFRONT, STORENAME is an optional argument defining storefront service store name. Applicable to STOREFRONT monitors.

storefrontacctservice

Enable/Disable probing for Account Service. Applicable only to Store Front monitors. For multi-tenancy configuration users my skip account service

Possible values: YES, NO

Default value: YES

netProfile

Name of the network profile.

originHost

Origin-Host value for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers.

originRealm

Origin-Realm value for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers.

hostIPAddress

Host-IP-Address value for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers. If Host-IP-Address is not specified, the appliance inserts the mapped IP (MIP) address or subnet IP (SNIP) address from which the CER request (the monitoring probe) is sent.

vendorId

Vendor-Id value for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers.

Minimum value: 0

productName

Product-Name value for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers.

firmwareRevision

Firmware-Revision value for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers.

Minimum value: 0

authApplicationId

List of Auth-Application-Id attribute value pairs (AVPs) for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers. A maximum of eight of these AVPs are supported in a monitoring CER message.

Minimum value: 0

Maximum value: 4294967295

acctApplicationId

List of Acct-Application-Id attribute value pairs (AVPs) for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers. A maximum of eight of these AVPs are supported in a monitoring message.

Minimum value: 0

Maximum value: 4294967295

inbandSecurityId

Inband-Security-Id for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers.

Possible values: NO_INBAND_SECURITY, TLS

supportedVendorIds

List of Supported-Vendor-Id attribute value pairs (AVPs) for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers. A maximum eight of these AVPs are supported in a monitoring message.

Minimum value: 1

Maximum value: 4294967295

vendorSpecificVendorId

Vendor-Id to use in the Vendor-Specific-Application-Id grouped attribute-value pair (AVP) in the monitoring CER message. To specify Auth-Application-Id or Acct-Application-Id in Vendor-Specific-Application-Id, use vendorSpecificAuthApplicationIds or vendorSpecificAcctApplicationIds, respectively. Only one Vendor-Id is supported for all the Vendor-Specific-Application-Id AVPs in a CER monitoring message.

Minimum value: 1

vendorSpecificAuthApplicationIds

List of Vendor-Specific-Auth-Application-Id attribute value pairs (AVPs) for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers. A maximum of eight of these AVPs are supported in a monitoring message. The specified value is combined with the value of vendorSpecificVendorId to obtain the Vendor-Specific-Application-Id AVP in the CER monitoring message.

Minimum value: 0

Maximum value: 4294967295

vendorSpecificAcctApplicationIds

List of Vendor-Specific-Acct-Application-Id attribute value pairs (AVPs) to use for monitoring Diameter servers. A maximum of eight of these AVPs are supported in a monitoring message. The specified value is combined with the value of vendorSpecificVendorId to obtain the Vendor-Specific-Application-Id AVP in the CER monitoring message.

Minimum value: 0

Maximum value: 4294967295

kcdAccount

KCD Account used by MSSQL monitor

storedb

Store the database list populated with the responses to monitor probes. Used in database specific load balancing if MSSQL-ECV/MYSQL-ECV monitor is configured.

Possible values: ENABLED, DISABLED

Default value: DISABLED

storefrontcheckbackendservices

This option will enable monitoring of services running on storefront server. Storefront services are monitored by probing to a Windows service that runs on the Storefront server and exposes details of which storefront services are running.

Possible values: YES, NO

Default value: NO

Example

```
add monitor http_mon http
```

rm lb monitor

Removes a monitor or a response code for an HTTP monitor. If you do not specify any response codes, the monitor is removed. If you provide any or all of the HTTP response codes that are configured for the monitor, only those specified response codes are removed; the monitor is not removed. Built-in monitors cannot be removed.

Synopsis

```
rm lb monitor <monitorName> <type> [-respCode <int[-int]> ...]
```

Arguments

monitorName

Name of the monitor.

type

Type of monitor that you want to create.

Possible values: PING, TCP, HTTP, TCP-ECV, HTTP-ECV, UDP-ECV, DNS, FTP, LDNS-PING, LDNS-TCP, LDNS-DNS, RADIUS, USER, HTTP-INLINE, SIP-UDP, LOAD, FTP-EXTENDED, SMTP, SNMP, NNTP, MYSQL, MYSQL-ECV, MSSQL-ECV, ORACLE-ECV, LDAP, POP3, CITRIX-XML-SERVICE, CITRIX-WEB-INTERFACE, DNS-TCP, RTSP, ARP, CITRIX-AG, CITRIX-AAC-LOGINPAGE, CITRIX-AAC-LAS, CITRIX-XD-DDC, ND6, CITRIX-WI-EXTENDED, DIAMETER, RADIUS_ACCOUNTING, STOREFRONT, APPC, CITRIX-XNC-ECV, CITRIX-XDM

respCode

Response codes to delete from the response code list configured for the HTTP monitor.

Example

```
rm monitor http_mon http
```

set lb monitor

Modifies the specified parameters of a monitor.

Synopsys

```
set lb monitor <monitorName> <type> [-action <action>] [-respCode <int[-int]> ...] [-httpRequest <string>] [-rtspRequest <string>] [-customHeaders <string>] [-maxForwards <positive_integer>] [-sipMethod <sipMethod>] [-sipregURI <string>] [-sipURI <string>] [-send <string>] [-recv <string>] [-query <string>] [-queryType <queryType>] [-userName <string>] [-password <string>] [-secondaryPassword <string>] [-logonpointName <string>] [-lasVersion <string>] [-radKey <string>] [-radNASid <string>] [-radNASip <ip_addr>] [-radAccountType <positive_integer>] [-radFramedIP <ip_addr>] [-radAPN <string>] [-radMSISDN <string>] [-radAccountSession <string>] [-LRTM ( ENABLED | DISABLED )] [-deviation <positive_integer> [<units>]] [-scriptName <string>] [-scriptArgs <string>] [-validateCred ( YES | NO )] [-domain <string>] [-dispatcherIP <ip_addr>] [-dispatcherPort <port>] [-interval <integer> [<units>]] [-resptimeout <integer> [<units>]] [-resptimeoutThresh <positive_integer>] [-retries <integer>] [-failureRetries <integer>] [-alertRetries <integer>] [-successRetries <integer>] [-downTime <integer> [<units>]] [-destIP <ip_addr|ipv6_addr>] [-destPort <port>] [-state ( ENABLED | DISABLED )] [-reverse ( YES | NO )] [-transparent ( YES | NO )] [-ipTunnel ( YES | NO )] [-tos ( YES | NO )] [-tosId <positive_integer>] [-secure ( YES | NO )] [-IPAddress <ip_addr|ipv6_addr|*> ...] [-group <string>] [-fileName <string>] [-baseDN <string>] [-bindDN <string>] [-filter <string>] [-attribute <string>] [-database <string>] [-oracleSid <string>] [-sqlQuery <text>] [-evalRule <expression>] [-snmpOID <string>] [-snmpCommunity <string>] [-snmpThreshold <string>] [-snmpVersion ( V1 | V2 )] [-metricTable <string>] [-metric <string>] [-metricThreshold <positive_integer>] [-metricWeight <positive_integer>] [-application <string>] [-sitePath <string>] [-storename <string>] [-storefrontacctservice ( YES | NO )] [-storefrontcheckbackendservices ( YES | NO )] [-netProfile <string>] [-mssqlProtocolVersion <mssqlProtocolVersion>] [-originHost <string>] [-originRealm <string>] [-hostIPAddress <ip_addr|ipv6_addr|*>] [-vendorId <positive_integer>] [-productName <string>] [-firmwareRevision <positive_integer>] [-authApplicationId <positive_integer> ...] [-acctApplicationId <positive_integer> ...] [-inbandSecurityId ( NO_INBAND_SECURITY | TLS )] [-supportedVendorIds <positive_integer> ...] [-vendorSpecificVendorId <positive_integer>] [-vendorSpecificAuthApplicationIds <positive_integer> ...] [-vendorSpecificAcctApplicationIds <positive_integer> ...] [-kcdAccount <string>]
```

Arguments

monitorName

Name of the monitor.

type

Type of monitor that you want to create.

Possible values: PING, TCP, HTTP, TCP-ECV, HTTP-ECV, UDP-ECV, DNS, FTP, LDNS-PING, LDNS-TCP, LDNS-DNS, RADIUS, USER, HTTP-INLINE, SIP-UDP, LOAD, FTP-EXTENDED, SMTP, SNMP, NNTP, MYSQL, MYSQL-ECV, MSSQL-ECV, ORACLE-ECV, LDAP, POP3, CITRIX-XML-SERVICE, CITRIX-WEB-INTERFACE, DNS-TCP, RTSP, ARP, CITRIX-AG, CITRIX-AAC-LOGINPAGE, CITRIX-AAC-LAS, CITRIX-XD-DDC, ND6, CITRIX-WI-EXTENDED, DIAMETER, RADIUS_ACCOUNTING, STOREFRONT, APPC, CITRIX-XNC-ECV, CITRIX-XDM

action

Action to perform when the response to an inline monitor (a monitor of type HTTP-INLINE) indicates that the service is down. A service monitored by an inline monitor is considered DOWN if the response code is not one of the codes that have been specified for the Response Code parameter.

Available settings function as follows:

- * NONE - Do not take any action. However, the show service command and the show lb monitor command indicate the total number of responses that were checked and the number of consecutive error responses received after the last successful probe.

- * LOG - Log the event in NSLOG or SYSLOG.

- * DOWN - Mark the service as being down, and then do not direct any traffic to the service until the configured down time has expired. Persistent connections to the service are terminated as soon as the service is marked as DOWN. Also, log the event in NSLOG or SYSLOG.

Possible values: NONE, LOG, DOWN

Default value: DOWN

respCode

Response codes for which to mark the service as UP. For any other response code, the action performed depends on the monitor type. HTTP monitors and RADIUS monitors mark the service as DOWN, while HTTP-INLINE monitors perform the action indicated by the Action parameter.

httpRequest

HTTP request to send to the server (for example, "HEAD /file.html").

rtspRequest

RTSP request to send to the server (for example, "OPTIONS *").

customHeaders

Custom header string to include in the monitoring probes.

maxForwards

Maximum number of hops that the SIP request used for monitoring can traverse to reach the server. Applicable only to monitors of type SIP-UDP.

Default value: 1

Minimum value: 0

Maximum value: 255

sipMethod

SIP method to use for the query. Applicable only to monitors of type SIP-UDP.

Possible values: OPTIONS, INVITE, REGISTER

sipregURI

SIP user to be registered. Applicable only if the monitor is of type SIP-UDP and the SIP Method parameter is set to REGISTER.

sipURI

SIP URI string to send to the service (for example, sip:sip.test). Applicable only to monitors of type SIP-UDP.

send

String to send to the service. Applicable to TCP-ECV, HTTP-ECV, and UDP-ECV monitors.

recv

String expected from the server for the service to be marked as UP. Applicable to TCP-ECV, HTTP-ECV, and UDP-ECV monitors.

query

Domain name to resolve as part of monitoring the DNS service (for example, example.com).

queryType

Type of DNS record for which to send monitoring queries. Set to Address for querying A records, AAAA for querying AAAA records, and Zone for querying the SOA record.

Possible values: Address, Zone, AAAA

userName

User name with which to probe the RADIUS, NNTP, FTP, FTP-EXTENDED, MYSQL, MSSQL, POP3, CITRIX-AG, CITRIX-XD-DDC, CITRIX-WI-EXTENDED, CITRIX-XNC or CITRIX-XDM server.

password

Password that is required for logging on to the RADIUS, NNTP, FTP, FTP-EXTENDED, MYSQL, MSSQL, POP3, CITRIX-AG, CITRIX-XD-DDC, CITRIX-WI-EXTENDED, CITRIX-XNC-ECV or CITRIX-XDM server. Used in conjunction with the user name specified for the User Name parameter.

secondaryPassword

Secondary password that users might have to provide to log on to the Access Gateway server. Applicable to CITRIX-AG monitors.

logonpointName

Name of the logon point that is configured for the Citrix Access Gateway Advanced Access Control software. Required if you want to monitor the associated login page or Logon Agent. Applicable to CITRIX-AAC-LAS and CITRIX-AAC-LOGINPAGE monitors.

lasVersion

Version number of the Citrix Advanced Access Control Logon Agent. Required by the CITRIX-AAC-LAS monitor.

radKey

Authentication key (shared secret text string) for RADIUS clients and servers to exchange. Applicable to monitors of type RADIUS and RADIUS_ACCOUNTING.

radNASid

NAS-Identifier to send in the Access-Request packet. Applicable to monitors of type RADIUS.

radNASip

Network Access Server (NAS) IP address to use as the source IP address when monitoring a RADIUS server. Applicable to monitors of type RADIUS and RADIUS_ACCOUNTING.

radAccountType

Account Type to be used in Account Request Packet. Applicable to monitors of type RADIUS_ACCOUNTING.

Default value: 1

Minimum value: 0

Maximum value: 15

radFramedIP

Source ip with which the packet will go out . Applicable to monitors of type RADIUS_ACCOUNTING.

radAPN

Called Station Id to be used in Account Request Packet. Applicable to monitors of type RADIUS_ACCOUNTING.

radMSISDN

Calling Stations Id to be used in Account Request Packet. Applicable to monitors of type RADIUS_ACCOUNTING.

radAccountSession

Account Session ID to be used in Account Request Packet. Applicable to monitors of type RADIUS_ACCOUNTING.

LRTM

Calculate the least response times for bound services. If this parameter is not enabled, the appliance does not learn the response times of the bound services. Also used for LRTM load balancing.

Possible values: ENABLED, DISABLED

deviation

Time value added to the learned average response time in dynamic response time monitoring (DRTM). When a deviation is specified, the appliance learns the average response time of bound services and adds the deviation to the average. The final value is then continually adjusted to accommodate response time variations over time. Specified in milliseconds, seconds, or minutes.

Minimum value: 0

Maximum value: 20939000

units

Unit of measurement for the Down Time parameter. Cannot be changed after the monitor is created.

Possible values: SEC, MSEC, MIN

Default value: SEC

scriptName

Path and name of the script to execute. The script must be available on the NetScaler appliance, in the /nsconfig/monitors/ directory.

scriptArgs

String of arguments for the script. The string is copied verbatim into the request.

validateCred

Validate the credentials of the Xen Desktop DDC server user. Applicable to monitors of type CITRIX-XD-DDC.

Possible values: YES, NO

Default value: NO

domain

Domain in which the XenDesktop Desktop Delivery Controller (DDC) servers or Web Interface servers are present. Required by CITRIX-XD-DDC and CITRIX-WI-EXTENDED monitors for logging on to the DDC servers and Web Interface servers, respectively.

dispatcherIP

IP address of the dispatcher to which to send the probe.

dispatcherPort

Port number on which the dispatcher listens for the monitoring probe.

interval

Time interval between two successive probes. Must be greater than the value of Response Time-out.

Default value: 5

Minimum value: 1

Maximum value: 20940000

resptimeout

Amount of time for which the appliance must wait before it marks a probe as FAILED. Must be less than the value specified for the Interval parameter.

Note: For UDP-ECV monitors for which a receive string is not configured, response timeout does not apply. For UDP-ECV monitors with no receive string, probe failure is indicated by an ICMP port unreachable error received from the service.

Default value: 2

Minimum value: 1

Maximum value: 20939000

resptimeoutThresh

Response time threshold, specified as a percentage of the Response Time-out parameter. If the response to a monitor probe has not arrived when the threshold is reached, the appliance generates an SNMP trap called monRespTimeoutAboveThresh. After the response time returns to a value below the threshold, the appliance generates a monRespTimeoutBelowThresh SNMP trap. For the traps to be generated, the "MONITOR-RTO-THRESHOLD" alarm must also be enabled.

Minimum value: 0

Maximum value: 100

retries

Maximum number of probes to send to establish the state of a service for which a monitoring probe failed.

Default value: 3

Minimum value: 1

Maximum value: 127

failureRetries

Number of retries that must fail, out of the number specified for the Retries parameter, for a service to be marked as DOWN. For example, if the Retries parameter is set to 10 and the Failure Retries parameter is set to 6, out of the ten probes sent, at least six probes must fail if the service is to be marked as DOWN. The default value of 0 means that all the retries must fail if the service is to be marked as DOWN.

Maximum value: 32

alertRetries

Number of consecutive probe failures after which the appliance generates an SNMP trap called monProbeFailed.

Maximum value: 32

successRetries

Number of consecutive successful probes required to transition a service's state from DOWN to UP.

Default value: 1

Minimum value: 1

Maximum value: 32

downTime

Time duration for which to wait before probing a service that has been marked as DOWN. Expressed in milliseconds, seconds, or minutes.

Default value: 30

Minimum value: 1

Maximum value: 20939000

destIP

IP address of the service to which to send probes. If the parameter is set to 0, the IP address of the server to which the monitor is bound is considered the destination IP address.

destPort

TCP or UDP port to which to send the probe. If the parameter is set to 0, the port number of the service to which the monitor is bound is considered the destination port. For a monitor of type USER, however, the destination port is the port number that is included in the HTTP request sent to the dispatcher. Does not apply to monitors of type PING.

state

State of the monitor. The DISABLED setting disables not only the monitor being configured, but all monitors of the same type, until the parameter is set to ENABLED. If the monitor is bound to a service, the state of the monitor is not taken into account when the state of the service is determined.

Possible values: ENABLED, DISABLED

Default value: ENABLED

reverse

Mark a service as DOWN, instead of UP, when probe criteria are satisfied, and as UP instead of DOWN when probe criteria are not satisfied.

Possible values: YES, NO

Default value: NO

transparent

The monitor is bound to a transparent device such as a firewall or router. The state of a transparent device depends on the responsiveness of the services behind it. If a transparent device is being monitored, a destination IP address must be specified. The probe is sent to the specified IP address by using the MAC address of the transparent device.

Possible values: YES, NO

Default value: NO

ipTunnel

Send the monitoring probe to the service through an IP tunnel. A destination IP address must be specified.

Possible values: YES, NO

Default value: NO

tos

Probe the service by encoding the destination IP address in the IP TOS (6) bits.

Possible values: YES, NO

tosId

The TOS ID of the specified destination IP. Applicable only when the TOS parameter is set.

Minimum value: 1

Maximum value: 63

secure

Use a secure SSL connection when monitoring a service. Applicable only to TCP based monitors. The secure option cannot be used with a CITRIX-AG monitor, because a CITRIX-AG monitor uses a secure connection by default.

Possible values: YES, NO

Default value: NO

IPAddress

Set of IP addresses expected in the monitoring response from the DNS server, if the record type is A or AAAA. Applicable to DNS monitors.

group

Name of a newsgroup available on the NNTP service that is to be monitored. The appliance periodically generates an NNTP query for the name of the newsgroup and evaluates the response. If the newsgroup is found on the server, the service is marked as UP. If the newsgroup does not exist or if the search fails, the service is marked as DOWN. Applicable to NNTP monitors.

fileName

Name of a file on the FTP server. The appliance monitors the FTP service by periodically checking the existence of the file on the server. Applicable to FTP-EXTENDED monitors.

baseDN

The base distinguished name of the LDAP service, from where the LDAP server can begin the search for the attributes in the monitoring query. Required for LDAP service monitoring.

bindDN

The distinguished name with which an LDAP monitor can perform the Bind operation on the LDAP server. Optional. Applicable to LDAP monitors.

filter

Filter criteria for the LDAP query. Optional.

attribute

Attribute to evaluate when the LDAP server responds to the query. Success or failure of the monitoring probe depends on whether the attribute exists in the response. Optional.

database

Name of the database to connect to during authentication.

oracleSid

Name of the service identifier that is used to connect to the Oracle database during authentication.

sqlQuery

SQL query for a MYSQL-ECV or MSSQL-ECV monitor. Sent to the database server after the server authenticates the connection.

evalRule

Default syntax expression that evaluates the database server's response to a MYSQL-ECV or MSSQL-ECV monitoring query. Must produce a Boolean result. The result determines the state of the server. If the expression returns TRUE, the probe succeeds.

For example, if you want the appliance to evaluate the error message to determine the state of the server, use the rule `MYSQL.RES.ROW(10).TEXT_ELEM(2).EQ("MySQL")`.

snmpOID

SNMP OID for SNMP monitors.

snmpCommunity

Community name for SNMP monitors.

snmpThreshold

Threshold for SNMP monitors.

snmpVersion

SNMP version to be used for SNMP monitors.

Possible values: V1, V2

metricTable

Metric table to which to bind metrics.

metric

Metric name in the metric table, whose setting is changed. A value zero disables the metric and it will not be used for load calculation

metricThreshold

Threshold to be used for that metric.

Minimum value: 0

metricWeight

The weight for the specified service metric with respect to others.

Minimum value: 1

Maximum value: 100

application

Name of the application used to determine the state of the service. Applicable to monitors of type CITRIX-XML-SERVICE.

sitePath

URL of the logon page. For monitors of type CITRIX-WEB-INTERFACE, to monitor a dynamic page under the site path, terminate the site path with a slash (/). Applicable to CITRIX-WEB-INTERFACE, CITRIX-WI-EXTENDED and CITRIX-XDM monitors.

storename

Store Name. For monitors of type STOREFRONT, STORENAME is an optional argument defining storefront service store name. Applicable to STOREFRONT monitors.

storefrontacctservice

Enable/Disable probing for Account Service. Applicable only to Store Front monitors. For multi-tenancy configuration users may skip account service

Possible values: YES, NO

Default value: YES

storefrontcheckbackendservices

This option will enable monitoring of services running on storefront server. Storefront services are monitored by probing to a Windows service that runs on the Storefront server and exposes details of which storefront services are running.

Possible values: YES, NO

Default value: NO

netProfile

Name of the network profile.

mssqlProtocolVersion

Version of MSSQL server that is to be monitored.

Possible values: 70, 2000, 2000SP1, 2005, 2008, 2008R2, 2012

Default value: 70

originHost

Origin-Host value for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers.

originRealm

Origin-Realm value for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers.

hostIPAddress

Host-IP-Address value for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers. If Host-IP-Address is not specified, the appliance inserts the mapped IP (MIP) address or subnet IP (SNIP) address from which the CER request (the monitoring probe) is sent.

vendorId

Vendor-Id value for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers.

Minimum value: 0

productName

Product-Name value for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers.

firmwareRevision

Firmware-Revision value for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers.

Minimum value: 0

authApplicationId

List of Auth-Application-Id attribute value pairs (AVPs) for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers. A maximum of eight of these AVPs are supported in a monitoring CER message.

Minimum value: 0

Maximum value: 4294967295

acctApplicationId

List of Acct-Application-Id attribute value pairs (AVPs) for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers. A maximum of eight of these AVPs are supported in a monitoring message.

Minimum value: 0

Maximum value: 4294967295

inbandSecurityId

Inband-Security-Id for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers.

Possible values: NO_INBAND_SECURITY, TLS

supportedVendorIds

List of Supported-Vendor-Id attribute value pairs (AVPs) for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers. A maximum eight of these AVPs are supported in a monitoring message.

Minimum value: 1

Maximum value: 4294967295

vendorSpecificVendorId

Vendor-Id to use in the Vendor-Specific-Application-Id grouped attribute-value pair (AVP) in the monitoring CER message. To specify Auth-Application-Id or Acct-Application-Id in Vendor-Specific-Application-Id, use vendorSpecificAuthApplicationIds or vendorSpecificAcctApplicationIds, respectively. Only one Vendor-Id is supported for all the Vendor-Specific-Application-Id AVPs in a CER monitoring message.

Minimum value: 1

vendorSpecificAuthApplicationIds

List of Vendor-Specific-Auth-Application-Id attribute value pairs (AVPs) for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers. A maximum of eight of these AVPs are supported in a monitoring message. The specified value is combined with the value of vendorSpecificVendorId to obtain the Vendor-Specific-Application-Id AVP in the CER monitoring message.

Minimum value: 0

Maximum value: 4294967295

vendorSpecificAcctApplicationIds

List of Vendor-Specific-Acct-Application-Id attribute value pairs (AVPs) to use for monitoring Diameter servers. A maximum of eight of these AVPs are supported in a monitoring message. The specified value is combined with the value of vendorSpecificVendorId to obtain the Vendor-Specific-Application-Id AVP in the CER monitoring message.

Minimum value: 0

Maximum value: 4294967295

kcdAccount

KCD Account used by MSSQL monitor

Example

```
set monitor http_mon http -respcode 100
```

unset lb monitor

Removes the specified parameter settings from the specified monitor. Attributes for which a default value is available revert to their default values. Refer to the set lb monitor command for meanings of the arguments.

Synopsys

```
unset lb monitor <monitorName> <type> [-IPAddress <ip_addr|ipv6_addr|*> ...] [-scriptName] [-destPort] [-netProfile] [-action] [-respCode] [-httpRequest] [-rtspRequest] [-customHeaders] [-maxForwards] [-sipMethod] [-sipregURI] [-send] [-recv] [-query] [-queryType] [-userName] [-password] [-secondaryPassword] [-logonpointName] [-lasVersion] [-radKey] [-radNASid] [-radNASip] [-radAccountType] [-radFramedIP] [-radAPN] [-radMSISDN] [-radAccountSession] [-LRTM] [-deviation] [-scriptArgs] [-validateCred] [-domain] [-dispatcherIP] [-dispatcherPort] [-interval] [-resptimeout] [-resptimeoutThresh] [-retries] [-failureRetries] [-alertRetries] [-successRetries] [-downTime] [-destIP] [-state] [-reverse] [-transparent] [-ipTunnel] [-tos] [-tosId] [-secure] [-group] [-fileName] [-baseDN] [-bindDN] [-filter] [-attribute] [-
```

database] [-oracleSid] [-sqlQuery] [-snmpOID] [-snmpCommunity] [-snmpThreshold] [-snmpVersion] [-metricTable] [-mssqlProtocolVersion] [-originHost] [-originRealm] [-hostIPAddress] [-vendorId] [-productName] [-firmwareRevision] [-authApplicationId] [-acctApplicationId] [-inbandSecurityId] [-supportedVendorIds] [-vendorSpecificVendorId] [-vendorSpecificAuthApplicationIds] [-vendorSpecificAcctApplicationIds] [-kcdAccount]

Example

```
set monitor dns_mon dns -ipaddress 10.102.27.230
```

enable lb monitor

Enable the monitor that is bound to a specific service. If no monitor name is specified, all monitors bound to the service are enabled.

Synopsys

```
enable lb monitor (<serviceName>@ | <serviceGroupName>@) [<monitorName>]
```

Arguments

serviceName

The name of the service to which the monitor is bound.

serviceGroupName

The name of the service group to which the monitor is to be bound.

monitorName

Name for the monitor. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

CLI Users: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my monitor" or 'my monitor').

Example

```
enable monitor http_svc http_mon To enable monitor for multiple services use the following
```

disable lb monitor

Disable the monitor for a service. If the monitor name is not specified, all monitors bound to the service are disabled.

Synopsys

```
disable lb monitor (<serviceName>@ | <serviceGroupName>@) [<monitorName>]
```

Arguments

serviceName

The name of the service being monitored.

serviceGroupName

The name of the service group being monitored.

monitorName

Name for the monitor. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

CLI Users: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my monitor" or 'my monitor').

Example

```
disable monitor http_svc http_mon
```

 To disable a monitor on multiple services use the following command:

bind lb monitor

Binds a monitor to a service or service group. Multiple monitors can be bound to a service or service group.

Synopsis

```
bind lb monitor <monitorName> [-state ( ENABLED | DISABLED )] [-weight <positive_integer>] [-state ( ENABLED | DISABLED )] [-weight <positive_integer>] [-metric <string> -metricThreshold <positive_integer> [-metricWeight <positive_integer>]]
```

Arguments

monitorName

Name of the monitor.

metric

Name of the metric to be polled by the monitor.

metricThreshold

Threshold for the specified metric. A value of zero disables the metric (the metric will not be used in load calculations).

Minimum value: 0

metricWeight

Weight to assign to the specified metric. A higher number specifies greater weight.

Default value: 1

Minimum value: 1

Maximum value: 100

Example

```
bind monitor http_mon http_svc
```

 To bind a monitor to multiple services use the following command:

unbind lb monitor

Unbinds a monitor from a service or service group.

Synopsis

```
unbind lb monitor <monitorName> -metric <string>
```

Arguments

monitorName

Name of the monitor.

metric

Name of the metric to be polled by the monitor.

Example

`unbind monitor http_mon http_svc` To unbind a monitor to multiple services use the follow:

show lb monitor

Displays the parameters of all the monitors configured on the appliance, or the parameters of the specified monitor.

Synopsys

`show lb monitor [<monitorName>] [<type>]` show lb monitor bindings - alias for 'show lb monbindings'

Arguments

monitorName

Name of the monitor.

type

Type of monitor that you want to create.

Possible values: PING, TCP, HTTP, TCP-ECV, HTTP-ECV, UDP-ECV, DNS, FTP, LDNS-PING, LDNS-TCP, LDNS-DNS, RADIUS, USER, HTTP-INLINE, SIP-UDP, LOAD, FTP-EXTENDED, SMTP, SNMP, NNTP, MYSQL, MYSQL-ECV, MSSQL-ECV, ORACLE-ECV, LDAP, POP3, CITRIX-XML-SERVICE, CITRIX-WEB-INTERFACE, DNS-TCP, RTSP, ARP, CITRIX-AG, CITRIX-AAC-LOGINPAGE, CITRIX-AAC-LAS, CITRIX-XD-DDC, ND6, CITRIX-WI-EXTENDED, DIAMETER, RADIUS_ACCOUNTING, STOREFRONT, APPC, CITRIX-XNC-ECV, CITRIX-XDM

Outputs

interval

The frequency at which the probe is sent to the service.

units

Giving the unit of the metric

resptimeout

The interval for which the system waits before it marks the probe as FAILED.

resptimeoutThresh

Response time threshold, specified as a percentage of the Response Time-out parameter. If the response to a monitor probe has not arrived when the threshold is reached, the appliance generates an SNMP trap called `monRespTimeoutAboveThresh`. After the response time returns to a value below the threshold, the appliance generates a `monRespTimeoutBelowThresh` SNMP trap. For the traps to be generated, the "MONITOR-RTO-THRESHOLD" alarm must also be enabled.

retries

Maximum number of probes to send to establish the state of a service for which a monitoring probe failed.

failureRetries

Number of retries that must fail, out of the number specified for the Retries parameter, for a service to be marked as DOWN. For example, if the Retries parameter is set to 10 and the Failure Retries parameter is set to 6, out of the ten probes sent, at least six probes must fail if the service is to be marked as DOWN. The default value of 0 means that all the retries must fail if the service is to be marked as DOWN.

alertRetries

The number of failures after which the system generates a SNMP trap.

successRetries

Number of consecutive successful probes required to transition a service's state from DOWN to UP.

downTime

The duration in seconds for which the system waits to make the next probe once the service is marked as DOWN.

destIP

The IP address to which the probe is sent.

destPort

The TCP/UDP port to which the probe is sent.

state

reverse

Mark a service as DOWN, instead of UP, when probe criteria are satisfied, and as UP instead of DOWN when probe criteria are not satisfied.

transparent

The state of the monitor for transparent devices.

ipTunnel

The state of the monitor for tunneled devices.

tos

TOS setting.

tosId

TOS ID

secure

The state of the secure monitoring of services.

action

Action to perform when the response to an inline monitor (a monitor of type HTTP-INLINE) indicates that the service is down. A service monitored by an inline monitor is considered DOWN if the response code is not one of the codes that have been specified for the Response Code parameter.

Available settings function as follows:

* NONE - Do not take any action. However, the show service command and the show lb monitor command indicate the total number of responses that were checked and the number of consecutive error responses received after the last successful probe.

* LOG - Log the event in NSLOG or SYSLOG.

* DOWN - Mark the service as being down, and then do not direct any traffic to the service until the configured down time has expired. Persistent connections to the service are terminated as soon as the service is marked as DOWN. Also, log the event in NSLOG or SYSLOG.

respCode

The response codes.

httpRequest

The HTTP request that is sent to the server.

rtspRequest

The RTSP request that is sent to the server.

send

The string that is sent to the service.

recv

The string that is expected from the server to mark the server as UP.

query

Domain name to resolve as part of monitoring the DNS service (for example, example.com).

queryType

Type of DNS record for which to send monitoring queries. Set to Address for querying A records, AAAA for querying AAAA records, and Zone for querying the SOA record.

userName

Username on the RADIUS/NNTP/FTP/FTP-EXTENDED/MYSQL/POP3/CITRIX-XD-DDC/CITRIX-WI-EXTENDED/CITRIX-XNC-ECV/CITRIX-XDM server. This user name is used in the probe.

password

Password used in RADIUS/NNTP/FTP/FTP-EXTENDED/MYSQL/POP3/LDAP/CITRIX-XD-DDC/CITRIX-WI-EXTENDED/CITRIX-XNC-ECV/CITRIX-XDM server monitoring.

secondaryPassword

Secondary password that users might have to provide to log on to the Access Gateway server. Applicable to CITRIX-AG monitors.

logonpointName

Logonpoint name used in Citrix AAC login page monitoring.

lasVersion

Version number of the Citrix Advanced Access Control Logon Agent. Required by the CITRIX-AAC-LAS monitor.

validateCred

Validate the credentials of the Xen Desktop DDC server user. Applicable to monitors of type CITRIX-XD-DDC.

domain

Domain in which the XenDesktop Desktop Delivery Controller (DDC) servers or Web Interface servers are present. Required by CITRIX-XD-DDC and CITRIX-WI-EXTENDED monitors for logging on to the DDC servers and Web Interface servers, respectively.

radKey

Authentication key (shared secret text string) for RADIUS clients and servers to exchange. Applicable to monitors of type RADIUS and RADIUS_ACCOUNTING.

radNASid

NAS-Identifier to send in the Access-Request packet. Applicable to monitors of type RADIUS.

radNASip

Network Access Server (NAS) IP address to use as the source IP address when monitoring a RADIUS server. Applicable to monitors of type RADIUS and RADIUS_ACCOUNTING.

radAccountType

Account Type to be used in Account Request Packet. Applicable to monitors of type RADIUS_ACCOUNTING.

radFramedIP

Source ip with which the packet will go out . Applicable to monitors of type RADIUS_ACCOUNTING.

radAPN

Called Station Id to be used in Account Request Packet. Applicable to monitors of type RADIUS_ACCOUNTING.

radMSISDN

Calling Stations Id to be used in Account Request Packet. Applicable to monitors of type RADIUS_ACCOUNTING.

radAccountSession

Account Session ID to be used in Account Request Packet. Applicable to monitors of type RADIUS_ACCOUNTING.

LRTM

Calculate the least response times for bound services. If this parameter is not enabled, the appliance does not learn the response times of the bound services. Also used for LRTM load balancing.

lrtmConf

State of LRTM configuration on the monitor.

lrtmConfStr

State of LRTM configuration on the monitor as STRING.

deviation

Deviation from the learnt response time for Dynamic Response Time monitoring.

dynamicResponseTimeout

Response timeout of the DRTM enabled monitor , calculated dynamically based on the history and current response time.

dynamicInterval

Interval between monitoring probes for DRTM enabled monitor , calculated dynamically based monitor response time.

scriptName

Path and name of the script to execute. The script must be available on the NetScaler appliance, in the /nsconfig/monitors/ directory.

scriptArgs

String of arguments for the script. The string is copied verbatim into the request.

dispatcherIP

IP address of the dispatcher to which to send the probe.

dispatcherPort

Port number on which the dispatcher listens for the monitoring probe.

sipURI

SIP URI string to send to the service (for example, sip:sip.test). Applicable only to monitors of type SIP-UDP.

sipMethod

Specifies SIP method to be used for the query

maxForwards

Maximum number of hops a sip monitor packet can go.

sipregURI

Specifies SIP user to be registered

customHeaders

The string that is sent to the service. Applicable to HTTP ,HTTP-ECV and RTSP monitor types.

IPAddress

Set of IP addresses expected in the monitoring response from the DNS server, if the record type is A or AAAA. Applicable to DNS monitors.

group

Name of a newsgroup available on the NNTP service that is to be monitored. The appliance periodically generates an NNTP query for the name of the newsgroup and evaluates the response. If the newsgroup is found on the server, the service is marked as UP. If the newsgroup does not exist or if the search fails, the service is marked as DOWN. Applicable to NNTP monitors.

fileName

Name of a file on the FTP server. The appliance monitors the FTP service by periodically checking the existence of the file on the server. Applicable to FTP-EXTENDED monitors.

baseDN

The base distinguished name of the LDAP service, from where the LDAP server can begin the search for the attributes in the monitoring query. Required for LDAP service monitoring.

bindDN

The distinguished name with which an LDAP monitor can perform the Bind operation on the LDAP server. Optional. Applicable to LDAP monitors.

filter

Filter criteria for the LDAP query. Optional.

attribute

Attribute to evaluate when the LDAP server responds to the query. Success or failure of the monitoring probe depends on whether the attribute exists in the response. Optional.

database

Name of the database to connect to during authentication.

oracleSid

Name of the service identifier that is used to connect to the Oracle database during authentication.

sqlQuery

SQL query for a MYSQL-ECV or MSSQL-ECV monitor. Sent to the database server after the server authenticates the connection.

evalRule

Default syntax expression that evaluates the database server's response to a MYSQL-ECV or MSSQL-ECV monitoring query. Must produce a Boolean result. The result determines the state of the server. If the expression returns TRUE, the probe succeeds.

For example, if you want the appliance to evaluate the error message to determine the state of the server, use the rule `MYSQL.RES.ROW(10).TEXT_ELEM(2).EQ("MySQL")`.

snmpOID

SNMP OID for SNMP monitors.

snmpCommunity

Community name for SNMP monitors.

snmpThreshold

Threshold for SNMP monitors.

snmpVersion

SNMP version to be used for SNMP monitoring.

metric

Metric name in the metric table, whose setting is changed

metricTable

Metric table, whose setting is changed

multimetricktable

Metric table to which to bind metrics, to be used only for output purposes.

metricThreshold

Threshold to be used for that metric.

metricWeight

The weight for the specified service metric with respect to others.

stateflag

Flags controlling the display.

flags

Used by build-in monitors.

application

Name of the application used to determine the state of the service. Applicable to monitors of type CITRIX-XML-SERVICE.

sitePath

URL of the logon page. For monitors of type CITRIX-WEB-INTERFACE, to monitor a dynamic page under the site path, terminate the site path with a slash (/). Applicable to CITRIX-WEB-INTERFACE, CITRIX-WI-EXTENDED and CITRIX-XDM monitors.

storename

Store Name. For monitors of type STOREFRONT, STORENAME is an optional argument defining storefront service store name. Applicable to STOREFRONT monitors.

storefrontacctservice

Enable/Disable probing for Account Service. Applicable only to Store Front monitors. For multi-tenancy configuration users my skip account service

storefrontcheckbackendservices

This option will enable monitoring of services running on storefront server. Storefront services are monitored by probing to a Windows service that runs on the Storefront server and exposes details of which storefront services are running.

hostName

Hostname in the FQDN format (Example: porche.cars.org). Applicable to STOREFRONT monitors.

netProfile

Name of the network profile.

mssqlProtocolVersion

Version of MSSQL server that is to be monitored.

originHost

Origin-Host value for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers.

originRealm

Origin-Realm value for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers.

hostIPAddress

Host-IP-Address value for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers. If Host-IP-Address is not specified, the appliance inserts the mapped IP (MIP) address or subnet IP (SNIP) address from which the CER request (the monitoring probe) is sent.

vendorId

Vendor-Id value for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers.

productName

Product-Name value for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers.

firmwareRevision

Firmware-Revision value for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers.

authApplicationId

List of Auth-Application-Id attribute value pairs (AVPs) for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers. A maximum of eight of these AVPs are supported in a monitoring CER message.

acctApplicationId

List of Acct-Application-Id attribute value pairs (AVPs) for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers. A maximum of eight of these AVPs are supported in a monitoring message.

inbandSecurityId

Inband-Security-Id for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers.

supportedVendorIds

List of Supported-Vendor-Id attribute value pairs (AVPs) for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers. A maximum eight of these AVPs are supported in a monitoring message.

vendorSpecificVendorId

Vendor-Id to use in the Vendor-Specific-Application-Id grouped attribute-value pair (AVP) in the monitoring CER message. To specify Auth-Application-Id or Acct-Application-Id in Vendor-Specific-Application-Id, use vendorSpecificAuthApplicationIds or vendorSpecificAcctApplicationIds, respectively. Only one Vendor-Id is supported for all the Vendor-Specific-Application-Id AVPs in a CER monitoring message.

vendorSpecificAuthApplicationIds

List of Vendor-Specific-Auth-Application-Id attribute value pairs (AVPs) for the Capabilities-Exchange-Request (CER) message to use for monitoring Diameter servers. A maximum of eight of these AVPs are supported in a monitoring message. The specified value is combined with the value of vendorSpecificVendorId to obtain the Vendor-Specific-Application-Id AVP in the CER monitoring message.

vendorSpecificAcctApplicationIds

List of Vendor-Specific-Acct-Application-Id attribute value pairs (AVPs) to use for monitoring Diameter servers. A maximum of eight of these AVPs are supported in a monitoring message. The specified value is combined with the value of vendorSpecificVendorId to obtain the Vendor-Specific-Application-Id AVP in the CER monitoring message.

serviceName**weight****serviceGroupName****kcdAccount**

KCD Account used by MSSQL monitor

storedb

Store the database list populated with the responses to monitor probes. Used in database specific load balancing if MSSQL-ECV/MYSQL-ECV monitor is configured.

devno**count**

Example

An example of the show monitor command output is as follows: 8 configured monitors: 1) 1

lb parameter

The following operations can be performed on "lb parameter":

[set](#) | [unset](#) | [show](#)

set lb parameter

Modifies the specified global load balancing parameters.

Synopsis

```
set lb parameter [-httpOnlyCookieFlag ( ENABLED | DISABLED )] [-useSecuredPersistenceCookie ( ENABLED |  
DISABLED )] [-cookiePassphrase ] [-consolidatedLConn ( YES | NO )] [-usePortForHashLb ( YES | NO )] [-  
preferDirectRoute ( YES | NO )] [-startupRRFactor <positive_integer>] [-monitorSkipMaxClient ( ENABLED |  
DISABLED )] [-monitorConnectionClose ( RESET | FIN )] [-vServerSpecificMac ( ENABLED | DISABLED )]
```

Arguments

httpOnlyCookieFlag

Include the HttpOnly attribute in persistence cookies. The HttpOnly attribute limits the scope of a cookie to HTTP requests and helps mitigate the risk of cross-site scripting attacks.

Possible values: ENABLED, DISABLED

Default value: ENABLED

useSecuredPersistenceCookie

Encode persistence cookie values using SHA2 hash.

Possible values: ENABLED, DISABLED

Default value: DISABLED

cookiePassphrase

Use this parameter to specify the passphrase used to generate secured persistence cookie value. It specifies the passphrase with a maximum of 31 characters.

consolidatedLConn

To find the service with the fewest connections, the virtual server uses the consolidated connection statistics from all the packet engines. The NO setting allows consideration of only the number of connections on the packet engine that received the new connection.

Possible values: YES, NO

Default value: YES

usePortForHashLb

Include the port number of the service when creating a hash for hash based load balancing methods. With the NO setting, only the IP address of the service is considered when creating a hash.

Possible values: YES, NO

Default value: YES

preferDirectRoute

Perform route lookup for traffic received by the NetScaler appliance, and forward the traffic according to configured routes. Do not set this parameter if you want a wildcard virtual server to direct packets received by the appliance to an intermediary device, such as a firewall, even if their destination is directly connected to the appliance. Route lookup is performed after the packets have been processed and returned by the intermediary device.

Possible values: YES, NO

Default value: YES

startupRRFactor

Number of requests, per service, for which to apply the round robin load balancing method before switching to the configured load balancing method, thus allowing services to ramp up gradually to full load. Until the specified number of requests is distributed, the NetScaler appliance is said to be implementing the slow start mode (or startup round robin). Implemented for a virtual server when one of the following is true:

- * The virtual server is newly created.
- * One or more services are newly bound to the virtual server.
- * One or more services bound to the virtual server are enabled.
- * The load balancing method is changed.

This parameter applies to all the load balancing virtual servers configured on the NetScaler appliance, except for those virtual servers for which the virtual server-level slow start parameters (New Service Startup Request Rate and Increment Interval) are configured. If the global slow start parameter and the slow start parameters for a given virtual server are not set, the appliance implements a default slow start for the virtual server, as follows:

- * For a newly configured virtual server, the appliance implements slow start for the first 100 requests received by the virtual server.
- * For an existing virtual server, if one or more services are newly bound or newly enabled, or if the load balancing method is changed, the appliance dynamically computes the number of requests for which to implement startup round robin. It obtains this number by multiplying the request rate by the number of bound services (it includes services that are marked as DOWN). For example, if the current request rate is 20 requests/s and ten services are bound to the virtual server, the appliance performs startup round robin for 200 requests.

Not applicable to a virtual server for which a hash based load balancing method is configured.

Minimum value: 0

monitorSkipMaxClient

When a monitor initiates a connection to a service, do not check to determine whether the number of connections to the service has reached the limit specified by the service's Max Clients setting. Enables monitoring to continue even if the service has reached its connection limit.

Possible values: ENABLED, DISABLED

Default value: DISABLED

monitorConnectionClose

Close monitoring connections by sending the service a connection termination message with the specified bit set.

Possible values: RESET, FIN

Default value: FIN

vServerSpecificMac

Allow a MAC-mode virtual server to accept traffic returned by an intermediary device, such as a firewall, to which the traffic was previously forwarded by another MAC-mode virtual server. The second virtual server can then distribute that traffic across the destination server farm. Also useful when load balancing Branch Repeater appliances.

Note: The second virtual server can also send the traffic to another set of intermediary devices, such as another set of firewalls. If necessary, you can configure multiple MAC-mode virtual servers to pass traffic successively through multiple sets of intermediary devices.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
set lb parameter -httponly (ENABLED|DISABLED)
```

unset lb parameter

Use this command to remove lb parameter settings. Refer to the set lb parameter command for meanings of the arguments.

Synopsys

```
unset lb parameter [-httpOnlyCookieFlag] [-useSecuredPersistenceCookie] [-cookiePassphrase] [-consolidatedLConn] [-usePortForHashLb] [-preferDirectRoute] [-startupRRFactor] [-monitorSkipMaxClient] [-monitorConnectionClose] [-vServerSpecificMac]
```

show lb parameter

Displays the global load balancing parameters.

Synopsys

```
show lb parameter
```

Outputs

httpOnlyCookieFlag

Include the HttpOnly attribute in persistence cookies. The HttpOnly attribute limits the scope of a cookie to HTTP requests and helps mitigate the risk of cross-site scripting attacks.

useSecuredPersistenceCookie

Encode persistence cookie values using SHA2 hash.

cookiePassphrase

Use this parameter to specify the passphrase used to generate secured persistence cookie value. It specifies the passphrase with a maximum of 31 characters.

consolidatedLConn

To find the service with the fewest connections, the virtual server uses the consolidated connection statistics from all the packet engines. The NO setting allows consideration of only the number of connections on the packet engine that received the new connection.

usePortForHashLb

Include the port number of the service when creating a hash for hash based load balancing methods. With the NO setting, only the IP address of the service is considered when creating a hash.

preferDirectRoute

Perform route lookup for traffic received by the NetScaler appliance, and forward the traffic according to configured routes. Do not set this parameter if you want a wildcard virtual server to direct packets received by the appliance to an intermediary device, such as a firewall, even if their destination is directly connected to the appliance. Route lookup is performed after the packets have been processed and returned by the intermediary device.

startupRRFactor

Used to change the factor of service hits after which vserver will come out of slowstart phase.

monitorSkipMaxClient

When a monitor initiates a connection to a service, do not check to determine whether the number of connections to the service has reached the limit specified by the service's Max Clients setting. Enables monitoring to continue even if the service has reached its connection limit.

monitorConnectionClose

Close monitoring connections by sending the service a connection termination message with the specified bit set.

vServerSpecificMac

Allow a MAC-mode virtual server to accept traffic returned by an intermediary device, such as a firewall, to which the traffic was previously forwarded by another MAC-mode virtual server. The second virtual server can then distribute that traffic across the destination server farm. Also useful when load balancing Branch Repeater appliances.

Note: The second virtual server can also send the traffic to another set of intermediary devices, such as another set of firewalls. If necessary, you can configure multiple MAC-mode virtual servers to pass traffic successively through multiple sets of intermediary devices.

sessionsThreshold

This option is used to get the upper-limit on the number of persistent sessions set by the administrator for this system

Example

```
show lb parameter
```

lb persistentSessions

The following operations can be performed on "lb persistentSessions":

[show](#) | [clear](#)

show lb persistentSessions

Get all vserver persistent sessions

Synopsys

show lb persistentSessions [<vServer>]

Arguments

vServer

The name of the virtual server.

Outputs

type

Type of Persistence.

typestring

Type of Persistence as String.

srcIP

SOURCE IP.

srcIPV6

SOURCE IPv6 ADDRESS.

destIP

DESTINATION IP.

destIPv6

DESTINATION IPv6 ADDRESS.

flags

IPv6 FLAGS.

destPort

Destination port.

vServerName

Virtual server name.

timeout

Persistent Session timeout.

referenceCount

Reference Count.

sipCallID

SIP CALLID.

persistenceParam

Specific persistence information . Callid in case of SIP_CALLID persistence entry , RTSP session id in case of RTSP_SESSIONID persistence entry.

cnamePersparam

The cname that is selected incase of gslb service

devno

count

stateflag

clear lb persistentSessions

Use this command to clear/flush persistent sessions

Synopsys

clear lb persistentSessions [<vServer>] [-persistenceParameter <string>]

Arguments

vServer

The name of the LB vserver whose persistence sessions are to be flushed. If not specified, all persistence sessions will be flushed .

persistenceParameter

The persistence parameter whose persistence sessions are to be flushed.

lb route

The following operations can be performed on "lb route":

[add](#) | [rm](#) | [show](#)

add lb route

Bind the route VIP to the route structure.

Synopsys

```
add lb route <network> <netmask> <gatewayName> [-td <positive_integer>]
```

Arguments

network

The IP address of the network to which the route belongs.

netmask

The netmask to which the route belongs.

gatewayName

The name of the route.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Default value: 0

Minimum value: 0

Maximum value: 4094

rm lb route

Remove the route VIP from the route structure.

Synopsys

```
rm lb route <network> <netmask> [-td <positive_integer>]
```

Arguments

network

The IP address of the network to which the route VIP belongs.

netmask

The netmask of the destination network.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Default value: 0

Minimum value: 0

Maximum value: 4094

show lb route

Display the names of the routes associated to the route structure using the `###add lb route###` command.

Synopsys

`show lb route [<network> <netmask> [-td <positive_integer>]]`

Arguments

network

The destination network or host.

netmask

The netmask of the destination network.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Default value: 0

Minimum value: 0

Maximum value: 4094

Outputs

gatewayName

The name of the route.

flags

State of the configured gateway.

devno

count

stateflag

lb route6

The following operations can be performed on "lb route6":

[add](#) | [rm](#) | [show](#)

add lb route6

Bind the route VIP to the route structure.

Synopsys

```
add lb route6 <network> <gatewayName> [-td <positive_integer>]
```

Arguments

network

The destination network.

gatewayName

The name of the route.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Default value: 0

Minimum value: 0

Maximum value: 4094

rm lb route6

Remove the route VIP from the route structure.

Synopsys

```
rm lb route6 <network> [-td <positive_integer>]
```

Arguments

network

The IP address of the network to which the route VIP belongs.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Default value: 0

Minimum value: 0

Maximum value: 4094

show lb route6

Display the names of the routes associated to the route structure using the `###add lb route6###` command.

Synopsys

show lb route6 [<network> [-td <positive_integer>]]

Arguments

network

The destination network or host.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Default value: 0

Minimum value: 0

Maximum value: 4094

Outputs

gatewayName

The name of the route.

flags

State of the configured gateway.

devno

count

stateflag

lb sipParameters

The following operations can be performed on "lb sipParameters":

[set](#) | [unset](#) | [show](#)

set lb sipParameters

Modifies the specified global SIP parameters.

Synopsis

```
set lb sipParameters [-rnatSrcPort <port>] [-rnatDstPort <port>] [-retryDur <integer>] [-addRportVip ( ENABLED | DISABLED )] [-sip503RateThreshold <positive_integer>]
```

Arguments

rnatSrcPort

Port number with which to match the source port in server-initiated SIP traffic. The rport parameter is added, without a value, to SIP packets that have a matching source port number, and CALL-ID based persistence is implemented for the responses received by the virtual server.

Default value: 0

rnatDstPort

Port number with which to match the destination port in server-initiated SIP traffic. The rport parameter is added, without a value, to SIP packets that have a matching source port number, and CALL-ID based persistence is implemented for the responses received by the virtual server.

Default value: 0

retryDur

Time, in seconds, for which a client must wait before initiating a connection after receiving a 503 Service Unavailable response from the SIP server. The time value is sent in the "Retry-After" header in the 503 response.

Default value: 120

Minimum value: 1

addRportVip

Add the rport parameter to the VIA headers of SIP requests that virtual servers receive from clients or servers.

Possible values: ENABLED, DISABLED

Default value: ENABLED

sip503RateThreshold

Maximum number of 503 Service Unavailable responses to generate, once every 10 milliseconds, when a SIP virtual server becomes unavailable.

Default value: 100

Minimum value: 0

Example

```
set sip parameter
```

unset lb sipParameters

Use this command to remove lb sipParameters settings. Refer to the set lb sipParameters command for meanings of the arguments.

Synopsys

```
unset lb sipParameters [-rnatSrcPort] [-rnatDstPort] [-retryDur] [-addRportVip] [-sip503RateThreshold]
```

show lb sipParameters

Displays the global SIP parameters.

Synopsys

```
show lb sipParameters
```

Outputs

rnatSrcPort

Port number with which to match the source port in server-initiated SIP traffic. The rport parameter is added, without a value, to SIP packets that have a matching source port number, and CALL-ID based persistence is implemented for the responses received by the virtual server.

rnatDstPort

Port number with which to match the destination port in server-initiated SIP traffic. The rport parameter is added, without a value, to SIP packets that have a matching source port number, and CALL-ID based persistence is implemented for the responses received by the virtual server.

retryDur

Time, in seconds, for which a client must wait before initiating a connection after receiving a 503 Service Unavailable response from the SIP server. The time value is sent in the "Retry-After" header in the 503 response.

addRportVip

Add the rport parameter to the VIA headers of SIP requests that virtual servers receive from clients or servers.

sip503RateThreshold

Maximum number of 503 Service Unavailable responses to generate, once every 10 milliseconds, when a SIP virtual server becomes unavailable.

Example

```
show sip parameter
```

lb vserver

The following operations can be performed on "lb vserver":

add | **rm** | **set** | **unset** | **bind** | **unbind** | **enable** | **disable** | **show** | **stat** | **rename**

add lb vserver

Creates a load balancing virtual server.

Synopsys

```
add lb vserver <name>@ <serviceType> [( <IPAddress>@ <port> [-range <positive_integer>] | (-IPPattern <ippat> -
IPMask <ipmask>)] [-persistenceType <persistenceType>] [-timeout <mins>] [-persistenceBackup ( SOURCEIP |
NONE )] [-backupPersistenceTimeout <mins>] [-lbMethod <lbMethod>] [-hashLength <positive_integer>] [-netmask
<netmask>] [-v6netmasklen <positive_integer>] [-dataLength <positive_integer>] [-dataOffset <positive_integer>]] [-
cookieName <string>] [-rule <expression>] [-Listenpolicy <expression>] [-Listenpriority <positive_integer>]] [-resRule
<expression>] [-persistMask <netmask>] [-v6persistmasklen <positive_integer>] [-pq ( ON | OFF )] [-sc ( ON | OFF )]
[-rtspNat ( ON | OFF )] [-m <m>] [-tosld <positive_integer>] [-sessionless ( ENABLED | DISABLED )] [-state (
ENABLED | DISABLED )] [-connfailover <connfailover>] [-redirectURL <URL>] [-cacheable ( YES | NO )] [-
cltTimeout <secs>] [-soMethod <soMethod>] [-soPersistence ( ENABLED | DISABLED )] [-soPersistenceTimeOut
<positive_integer>] [-healthThreshold <positive_integer>] [-soThreshold <positive_integer>] [-soBackupAction
<soBackupAction>] [-redirectPortRewrite ( ENABLED | DISABLED )] [-downStateFlush ( ENABLED | DISABLED )] [-
backupVServer <string>] [-disablePrimaryOnDown ( ENABLED | DISABLED )] [-insertVserverIPPort
<insertVserverIPPort> <vipHeader>] [-AuthenticationHost <string>] [-Authentication ( ON | OFF )] [-authn401 ( ON
| OFF )] [-authnVsName <string>] [-push ( ENABLED | DISABLED )] [-pushVserver <string>] [-pushLabel
<expression>] [-pushMultiClients ( YES | NO )] [-tcpProfileName <string>] [-httpProfileName <string>] [-
dbProfileName <string>] [-comment <string>] [-l2Conn ( ON | OFF )] [-oracleServerVersion ( 10G | 11G )] [-
mssqlServerVersion <mssqlServerVersion>] [-mysqlProtocolVersion <positive_integer>] [-mysqlServerVersion
<string>] [-mysqlCharacterSet <positive_integer>] [-mysqlServerCapabilities <positive_integer>] [-appflowLog (
ENABLED | DISABLED )] [-netProfile <string>] [-icmpVsrResponse ( PASSIVE | ACTIVE )] [-RHlstate ( PASSIVE |
ACTIVE )] [-newServiceRequest <positive_integer>] [-<newServiceRequestUnit>]] [-
newServiceRequestIncrementInterval <positive_integer>] [-minAutoscaleMembers <positive_integer>] [-
maxAutoscaleMembers <positive_integer>] [-persistAVPno <positive_integer> ...] [-skippersistency
<skippersistency>] [-td <positive_integer>] [-authnProfile <string>] [-macmodeRetainvlan ( ENABLED | DISABLED )]
[-dbsLb ( ENABLED | DISABLED )] [-dns64 ( ENABLED | DISABLED )] [-bypassAAAA ( YES | NO )] [-
RecursionAvailable ( YES | NO )] [-processLocal ( ENABLED | DISABLED )]
```

Arguments

name

Name for the virtual server. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at sign (@), equal sign (=), and hyphen (-) characters. Can be changed after the virtual server is created.

CLI Users: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my vserver" or 'my vserver').

serviceType

Protocol used by the service (also called the service type).

Possible values: HTTP, FTP, TCP, UDP, SSL, SSL_BRIDGE, SSL_TCP, DTLS, NNTP, DNS, DHCPRA, ANY, SIP_UDP, DNS_TCP, RTSP, PUSH, SSL_PUSH, RADIUS, RDP, MYSQL, MSSQL, DIAMETER, SSL_DIAMETER, FTTP, ORACLE

IPAddress

IPv4 or IPv6 address to assign to the virtual server.

IPPattern

IP address pattern, in dotted decimal notation, for identifying packets to be accepted by the virtual server. The IP Mask parameter specifies which part of the destination IP address is matched against the pattern. Mutually exclusive with the IP Address parameter.

For example, if the IP pattern assigned to the virtual server is 198.51.100.0 and the IP mask is 255.255.240.0 (a forward mask), the first 20 bits in the destination IP addresses are matched with the first 20 bits in the pattern. The virtual server accepts requests with IP addresses that range from 198.51.96.1 to 198.51.111.254. You can also use a pattern such as 0.0.2.2 and a mask such as 0.0.255.255 (a reverse mask).

If a destination IP address matches more than one IP pattern, the pattern with the longest match is selected, and the associated virtual server processes the request. For example, if virtual servers vs1 and vs2 have the same IP pattern, 0.0.100.128, but different IP masks of 0.0.255.255 and 0.0.224.255, a destination IP address of 198.51.100.128 has the longest match with the IP pattern of vs1. If a destination IP address matches two or more virtual servers to the same extent, the request is processed by the virtual server whose port number matches the port number in the request.

IPMask

IP mask, in dotted decimal notation, for the IP Pattern parameter. Can have leading or trailing non-zero octets (for example, 255.255.240.0 or 0.0.255.255). Accordingly, the mask specifies whether the first n bits or the last n bits of the destination IP address in a client request are to be matched with the corresponding bits in the IP pattern. The former is called a forward mask. The latter is called a reverse mask.

port

Port number for the virtual server.

range

Number of IP addresses that the appliance must generate and assign to the virtual server. The virtual server then functions as a network virtual server, accepting traffic on any of the generated IP addresses. The IP addresses are generated automatically, as follows:

- * For a range of n, the last octet of the address specified by the IP Address parameter increments n-1 times.
- * If the last octet exceeds 255, it rolls over to 0 and the third octet increments by 1.

Note: The Range parameter assigns multiple IP addresses to one virtual server. To generate an array of virtual servers, each of which owns only one IP address, use brackets in the IP Address and Name parameters to specify the range. For example:

```
add lb vserver my_vserver[1-3] HTTP 192.0.2.[1-3] 80
```

Default value: 1

Minimum value: 1

Maximum value: 254

persistenceType

Type of persistence for the virtual server. Available settings function as follows:

- * SOURCEIP - Connections from the same client IP address belong to the same persistence session.
- * COOKIEINSERT - Connections that have the same HTTP Cookie, inserted by a Set-Cookie directive from a server, belong to the same persistence session.
- * SSLSESSION - Connections that have the same SSL Session ID belong to the same persistence session.
- * CUSTOMSERVERID - Connections with the same server ID form part of the same session. For this persistence type, set the Server ID (CustomServerID) parameter for each service and configure the Rule parameter to identify the server ID in a request.
- * RULE - All connections that match a user defined rule belong to the same persistence session.
- * URLPASSIVE - Requests that have the same server ID in the URL query belong to the same persistence session. The server ID is the hexadecimal representation of the IP address and port of the service to which the request must be forwarded. This persistence type requires a rule to identify the server ID in the request.
- * DESTIP - Connections to the same destination IP address belong to the same persistence session.

* SRCIPDESTIP - Connections that have the same source IP address and destination IP address belong to the same persistence session.

* CALLID - Connections that have the same CALL-ID SIP header belong to the same persistence session.

* RTSPSID - Connections that have the same RTSP Session ID belong to the same persistence session.

Possible values: SOURCEIP, COOKIEINSERT, SSLSESSION, RULE, URLPASSIVE, CUSTOMSERVERID, DESTIP, SRCIPDESTIP, CALLID, RTSPSID, DIAMETER, NONE

timeout

Time period for which a persistence session is in effect.

Default value: 2

Maximum value: 1440

persistenceBackup

Backup persistence type for the virtual server. Becomes operational if the primary persistence mechanism fails.

Possible values: SOURCEIP, NONE

backupPersistenceTimeout

Time period for which backup persistence is in effect.

Default value: 2

Minimum value: 2

Maximum value: 1440

lbMethod

Load balancing method. The available settings function as follows:

* ROUNDROBIN - Distribute requests in rotation, regardless of the load. Weights can be assigned to services to enforce weighted round robin distribution.

* LEASTCONNECTION (default) - Select the service with the fewest connections.

* LEASTRESPONSETIME - Select the service with the lowest average response time.

* LEASTBANDWIDTH - Select the service currently handling the least traffic.

* LEASTPACKETS - Select the service currently serving the lowest number of packets per second.

* CUSTOMLOAD - Base service selection on the SNMP metrics obtained by custom load monitors.

* LRTM - Select the service with the lowest response time. Response times are learned through monitoring probes. This method also takes the number of active connections into account.

Also available are a number of hashing methods, in which the appliance extracts a predetermined portion of the request, creates a hash of the portion, and then checks whether any previous requests had the same hash value. If it finds a match, it forwards the request to the service that served those previous requests. Following are the hashing methods:

* URLHASH - Create a hash of the request URL (or part of the URL).

* DOMAINHASH - Create a hash of the domain name in the request (or part of the domain name). The domain name is taken from either the URL or the Host header. If the domain name appears in both locations, the URL is preferred. If the request does not contain a domain name, the load balancing method defaults to LEASTCONNECTION.

* DESTINATIONIPHASH - Create a hash of the destination IP address in the IP header.

* SOURCEIPHASH - Create a hash of the source IP address in the IP header.

* TOKEN - Extract a token from the request, create a hash of the token, and then select the service to which any previous requests with the same token hash value were sent.

* SRCIPDESTIPHASH - Create a hash of the string obtained by concatenating the source IP address and destination IP address in the IP header.

* SRCIPSRCPORHASH - Create a hash of the source IP address and source port in the IP header.

* CALLIDHASH - Create a hash of the SIP Call-ID header.

Possible values: ROUNDROBIN, LEASTCONNECTION, LEASTRESPONSETIME, URLHASH, DOMAINHASH, DESTINATIONIPHASH, SOURCEIPHASH, SRCIPDESTIPHASH, LEASTBANDWIDTH, LEASTPACKETS, TOKEN, SRCIPSRCPORHASH, LRTM, CALLIDHASH, CUSTOMLOAD, LEASTREQUEST

Default value: LEASTCONNECTION

hashLength

Number of bytes to consider for the hash value used in the URLHASH and DOMAINHASH load balancing methods.

Default value: 80

Minimum value: 1

Maximum value: 4096

netmask

IPv4 subnet mask to apply to the destination IP address or source IP address when the load balancing method is DESTINATIONIPHASH or SOURCEIPHASH.

Default value: 0xFFFFFFFF

v6netmasklen

Number of bits to consider in an IPv6 destination or source IP address, for creating the hash that is required by the DESTINATIONIPHASH and SOURCEIPHASH load balancing methods.

Default value: 128

Minimum value: 1

Maximum value: 128

cookieName

Use this parameter to specify the cookie name for COOKIE persistence type. It specifies the name of cookie with a maximum of 32 characters. If not specified, cookie name is internally generated.

rule

Expression, or name of a named expression, against which traffic is evaluated. Written in the classic or default syntax.

Note:

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

The following requirements apply only to the NetScaler CLI:

* If the expression includes one or more spaces, enclose the entire expression in double quotation marks.

* If the expression itself includes double quotation marks, escape the quotations by using the \ character.

* Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

Default value: "none"

Listenpolicy

Default syntax expression identifying traffic accepted by the virtual server. Can be either an expression (for example, CLIENT.IP.DST.IN_SUBNET(192.0.2.0/24) or the name of a named expression. In the above example, the virtual server accepts all requests whose destination IP address is in the 192.0.2.0/24 subnet.

Default value: "none"

Listenpriority

Integer specifying the priority of the listen policy. A higher number specifies a lower priority. If a request matches the listen policies of more than one virtual server the virtual server whose listen policy has the highest priority (the lowest priority number) accepts the request.

Default value: 101

Minimum value: 0

Maximum value: 101

resRule

Default syntax expression specifying which part of a server's response to use for creating rule based persistence sessions (persistence type RULE). Can be either an expression or the name of a named expression.

Example:

HTTP.RES.HEADER("setcookie").VALUE(0).TYPECAST_NVLIST_T('=';';').VALUE("server1").

Default value: "none"

persistMask

Persistence mask for IP based persistence types, for IPv4 virtual servers.

Default value: 0xFFFFFFFF

v6persistmasklen

Persistence mask for IP based persistence types, for IPv6 virtual servers.

Default value: 128

Minimum value: 1

Maximum value: 128

pq

Use priority queuing on the virtual server. based persistence types, for IPv6 virtual servers.

Possible values: ON, OFF

Default value: OFF

sc

Use SureConnect on the virtual server.

Possible values: ON, OFF

Default value: OFF

rtspNat

Use network address translation (NAT) for RTSP data connections.

Possible values: ON, OFF

Default value: OFF

m

Redirection mode for load balancing. Available settings function as follows:

- * IP - Before forwarding a request to a server, change the destination IP address to the server's IP address.
- * MAC - Before forwarding a request to a server, change the destination MAC address to the server's MAC address. The destination IP address is not changed. MAC-based redirection mode is used mostly in firewall load balancing deployments.
- * IPTUNNEL - Perform IP-in-IP encapsulation for client IP packets. In the outer IP headers, set the destination IP address to the IP address of the server and the source IP address to the subnet IP (SNIP). The client IP packets are not modified. Applicable to both IPv4 and IPv6 packets.
- * TOS - Encode the virtual server's TOS ID in the TOS field of the IP header.

You can use either the IPTUNNEL or the TOS option to implement Direct Server Return (DSR).

Possible values: IP, MAC, IPTUNNEL, TOS

Default value: IP

tosId

TOS ID of the virtual server. Applicable only when the load balancing redirection mode is set to TOS.

Minimum value: 1

Maximum value: 63

dataLength

Length of the token to be extracted from the data segment of an incoming packet, for use in the token method of load balancing. The length of the token, specified in bytes, must not be greater than 24 KB. Applicable to virtual servers of type TCP.

Minimum value: 1

Maximum value: 100

dataOffset

Offset to be considered when extracting a token from the TCP payload. Applicable to virtual servers, of type TCP, using the token method of load balancing. Must be within the first 24 KB of the TCP payload.

Minimum value: 0

Maximum value: 25400

sessionless

Perform load balancing on a per-packet basis, without establishing sessions. Recommended for load balancing of intrusion detection system (IDS) servers and scenarios involving direct server return (DSR), where session information is unnecessary.

Possible values: ENABLED, DISABLED

Default value: DISABLED

state

State of the load balancing virtual server.

Possible values: ENABLED, DISABLED

Default value: ENABLED

connfailover

Mode in which the connection failover feature must operate for the virtual server. After a failover, established TCP connections and UDP packet flows are kept active and resumed on the secondary appliance. Clients remain connected to the same servers. Available settings function as follows:

- * STATEFUL - The primary appliance shares state information with the secondary appliance, in real time, resulting in some runtime processing overhead.

- * STATELESS - State information is not shared, and the new primary appliance tries to re-create the packet flow on the basis of the information contained in the packets it receives.

- * DISABLED - Connection failover does not occur.

Possible values: DISABLED, STATEFUL, STATELESS

Default value: DISABLED

redirectURL

URL to which to redirect traffic if the virtual server becomes unavailable.

WARNING! Make sure that the domain in the URL does not match the domain specified for a content switching policy. If it does, requests are continuously redirected to the unavailable virtual server.

cacheable

Route cacheable requests to a cache redirection virtual server. The load balancing virtual server can forward requests only to a transparent cache redirection virtual server that has an IP address and port combination of *:80, so such a cache redirection virtual server must be configured on the appliance.

Possible values: YES, NO

Default value: NO

cltTimeout

Idle time, in seconds, after which a client connection is terminated.

Default value: -1

Maximum value: 31536000

soMethod

Type of threshold that, when exceeded, triggers spillover. Available settings function as follows:

- * CONNECTION - Spillover occurs when the number of client connections exceeds the threshold.

- * DYNAMICCONNECTION - Spillover occurs when the number of client connections at the virtual server exceeds the sum of the maximum client (Max Clients) settings for bound services. Do not specify a spillover threshold for this setting, because the threshold is implied by the Max Clients settings of bound services.

- * BANDWIDTH - Spillover occurs when the bandwidth consumed by the virtual server's incoming and outgoing traffic exceeds the threshold.

- * HEALTH - Spillover occurs when the percentage of weights of the services that are UP drops below the threshold. For example, if services svc1, svc2, and svc3 are bound to a virtual server, with weights 1, 2, and 3, and the spillover threshold is 50%, spillover occurs if svc1 and svc3 or svc2 and svc3 transition to DOWN.

- * NONE - Spillover does not occur.

Possible values: CONNECTION, DYNAMICCONNECTION, BANDWIDTH, HEALTH, NONE

soPersistence

If spillover occurs, maintain source IP address based persistence for both primary and backup virtual servers.

Possible values: ENABLED, DISABLED

Default value: DISABLED

soPersistenceTimeOut

Timeout for spillover persistence, in minutes.

Default value: 2

Minimum value: 2

Maximum value: 1440

healthThreshold

Threshold in percent of active services below which vserver state is made down. If this threshold is 0, vserver state will be up even if one bound service is up.

Default value: 0

Minimum value: 0

Maximum value: 100

soThreshold

Threshold at which spillover occurs. Specify an integer for the CONNECTION spillover method, a bandwidth value in kilobits per second for the BANDWIDTH method (do not enter the units), or a percentage for the HEALTH method (do not enter the percentage symbol).

Minimum value: 1

Maximum value: 4294967287

soBackupAction

Action to be performed if spillover is to take effect, but no backup chain to spillover is usable or exists

Possible values: DROP, ACCEPT, REDIRECT

redirectPortRewrite

Rewrite the port and change the protocol to ensure successful HTTP redirects from services.

Possible values: ENABLED, DISABLED

Default value: DISABLED

downStateFlush

Flush all active transactions associated with a virtual server whose state transitions from UP to DOWN. Do not enable this option for applications that must complete their transactions.

Possible values: ENABLED, DISABLED

Default value: ENABLED

backupVServer

Name of the backup virtual server to which to forward requests if the primary virtual server goes DOWN or reaches its spillover threshold.

disablePrimaryOnDown

If the primary virtual server goes down, do not allow it to return to primary status until manually enabled.

Possible values: ENABLED, DISABLED

Default value: DISABLED

insertVserverIPPort

Insert an HTTP header, whose value is the IP address and port number of the virtual server, before forwarding a request to the server. The format of the header is <vipHeader>: <virtual server IP address>_<port number>, where vipHeader is the name that you specify for the header. If the virtual server has an IPv6 address, the address in the header is enclosed in brackets ([and]) to separate it from the port

number. If you have mapped an IPv4 address to a virtual server's IPv6 address, the value of this parameter determines which IP address is inserted in the header, as follows:

* VIPADDR - Insert the IP address of the virtual server in the HTTP header regardless of whether the virtual server has an IPv4 address or an IPv6 address. A mapped IPv4 address, if configured, is ignored.

* V6TOV4MAPPING - Insert the IPv4 address that is mapped to the virtual server's IPv6 address. If a mapped IPv4 address is not configured, insert the IPv6 address.

* OFF - Disable header insertion.

Possible values: OFF, VIPADDR, V6TOV4MAPPING

vipHeader

Name for the inserted header. The default name is vip-header.

AuthenticationHost

Fully qualified domain name (FQDN) of the authentication virtual server to which the user must be redirected for authentication. Make sure that the Authentication parameter is set to ENABLED.

Authentication

Enable or disable user authentication.

Possible values: ON, OFF

Default value: OFF

authn401

Enable or disable user authentication with HTTP 401 responses.

Possible values: ON, OFF

Default value: OFF

authnVsName

Name of an authentication virtual server with which to authenticate users.

push

Process traffic with the push virtual server that is bound to this load balancing virtual server.

Possible values: ENABLED, DISABLED

Default value: DISABLED

pushVserver

Name of the load balancing virtual server, of type PUSH or SSL_PUSH, to which the server pushes updates received on the load balancing virtual server that you are configuring.

pushLabel

Expression for extracting a label from the server's response. Can be either an expression or the name of a named expression.

Default value: "none"

pushMultiClients

Allow multiple Web 2.0 connections from the same client to connect to the virtual server and expect updates.

Possible values: YES, NO

Default value: NO

tcpProfileName

Name of the TCP profile whose settings are to be applied to the virtual server.

httpProfileName

Name of the HTTP profile whose settings are to be applied to the virtual server.

dbProfileName

Name of the DB profile whose settings are to be applied to the virtual server.

comment

Any comments that you might want to associate with the virtual server.

I2Conn

Use Layer 2 parameters (channel number, MAC address, and VLAN ID) in addition to the 4-tuple (<source IP>:<source port>:<destination IP>:<destination port>) that is used to identify a connection. Allows multiple TCP and non-TCP connections with the same 4-tuple to co-exist on the NetScaler appliance.

Possible values: ON, OFF

oracleServerVersion

Oracle server version

Possible values: 10G, 11G

Default value: 10G

mssqlServerVersion

For a load balancing virtual server of type MSSQL, the Microsoft SQL Server version. Set this parameter if you expect some clients to run a version different from the version of the database. This setting provides compatibility between the client-side and server-side connections by ensuring that all communication conforms to the server's version.

Possible values: 70, 2000, 2000SP1, 2005, 2008, 2008R2, 2012

Default value: 2008R2

mysqlProtocolVersion

MySQL protocol version that the virtual server advertises to clients.

Default value: NSA_MYSQL_PROTOCOL_VER_DEFAULT

Minimum value: 0

mysqlServerVersion

MySQL server version string that the virtual server advertises to clients.

Default value: NSA_MYSQL_SERVER_VER_DEFAULT

mysqlCharacterSet

Character set that the virtual server advertises to clients.

Default value: NSA_MYSQL_CHAR_SET_DEFAULT

Minimum value: 0

mysqlServerCapabilities

Server capabilities that the virtual server advertises to clients.

Default value: NSA_MYSQL_SVR_CAPABILITIES_DEFAULT

Minimum value: 0

appflowLog

Apply AppFlow logging to the virtual server.

Possible values: ENABLED, DISABLED

Default value: ENABLED

netProfile

Name of the network profile to associate with the virtual server. If you set this parameter, the virtual server uses only the IP addresses in the network profile as source IP addresses when initiating connections with servers.

icmpVsrResponse

How the NetScaler appliance responds to ping requests received for an IP address that is common to one or more virtual servers. Available settings function as follows:

- * If set to PASSIVE on all the virtual servers that share the IP address, the appliance always responds to the ping requests.

- * If set to ACTIVE on all the virtual servers that share the IP address, the appliance responds to the ping requests if at least one of the virtual servers is UP. Otherwise, the appliance does not respond.

- * If set to ACTIVE on some virtual servers and PASSIVE on the others, the appliance responds if at least one virtual server with the ACTIVE setting is UP. Otherwise, the appliance does not respond.

Note: This parameter is available at the virtual server level. A similar parameter, ICMP Response, is available at the IP address level, for IPv4 addresses of type VIP. To set that parameter, use the add ip command in the CLI or the Create IP dialog box in the GUI.

Possible values: PASSIVE, ACTIVE

Default value: PASSIVE

RHIstate

Route Health Injection (RHI) functionality of the NetScaler appliance for advertising the route of the VIP address associated with the virtual server. When Vserver RHI Level (RHI) parameter is set to VSVR_CNTRL, the following are different RHI behaviors for the VIP address on the basis of RHIstate (RHI STATE) settings on the virtual servers associated with the VIP address:

- * If you set RHI STATE to PASSIVE on all virtual servers, the NetScaler ADC always advertises the route for the VIP address.

- * If you set RHI STATE to ACTIVE on all virtual servers, the NetScaler ADC advertises the route for the VIP address if at least one of the associated virtual servers is in UP state.

- * If you set RHI STATE to ACTIVE on some and PASSIVE on others, the NetScaler ADC advertises the route for the VIP address if at least one of the associated virtual servers, whose RHI STATE set to ACTIVE, is in UP state.

Possible values: PASSIVE, ACTIVE

Default value: PASSIVE

newServiceRequest

Number of requests, or percentage of the load on existing services, by which to increase the load on a new service at each interval in slow-start mode. A non-zero value indicates that slow-start is applicable. A zero value indicates that the global RR startup parameter is applied. Changing the value to zero will cause services currently in slow start to take the full traffic as determined by the LB method. Subsequently, any new services added will use the global RR factor.

Default value: 0

Minimum value: 0

newServiceRequestUnit

Units in which to increment load at each interval in slow-start mode.

Possible values: PER_SECOND, PERCENT

Default value: PER_SECOND

newServiceRequestIncrementInterval

Interval, in seconds, between successive increments in the load on a new service or a service whose state has just changed from DOWN to UP. A value of 0 (zero) specifies manual slow start.

Default value: 0

Minimum value: 0

Maximum value: 3600

minAutoscaleMembers

Minimum number of members expected to be present when vservers are used in Autoscale.

Default value: 0

Minimum value: 0

Maximum value: 5000

maxAutoscaleMembers

Maximum number of members expected to be present when vservers are used in Autoscale.

Default value: 0

Minimum value: 0

Maximum value: 5000

persistAVPno

Persist AVP number for Diameter Persistency.

In case this AVP is not defined in Base RFC 3588 and it is nested inside a Grouped AVP,

define a sequence of AVP numbers (max 3) in order of parent to child. So say persist AVP number X

is nested inside AVP Y which is nested in Z, then define the list as Z Y X

Minimum value: 1

skippersistency

This argument decides the behavior incase the service which is selected from an existing persistence session has reached threshold.

Possible values: Bypass, ReLb, None

Default value: None

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

authnProfile

Name of the authentication profile to be used when authentication is turned on.

macmodeRetainvlan

This option is used to retain vlan information of incoming packet when macmode is enabled

Possible values: ENABLED, DISABLED

Default value: DISABLED

lbsLb

Enable database specific load balancing for MySQL and MSSQL service types.

Possible values: ENABLED, DISABLED

Default value: DISABLED

dns64

This argument is for enabling/disabling the dns64 on lbvserver

Possible values: ENABLED, DISABLED

bypassAAAA

If this option is enabled while resolving DNS64 query AAAA queries are not sent to back end dns server

Possible values: YES, NO

Default value: NO

RecursionAvailable

When set to YES, this option causes the DNS replies from this vserver to have the RA bit turned on. Typically one would set this option to YES, when the vserver is load balancing a set of DNS servers that support recursive queries.

Possible values: YES, NO

Default value: NO

processLocal

By turning on this option packets destined to a vserver in a cluster will not undergo any steering. Turn this option for single packet request response mode or when the upstream device is performing a proper RSS for connection based distribution.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
add lb vserver http_vsvr http 10.102.1.10 80 To add multiple vservers at once use the fo:
```

rm lb vserver

Removes a virtual server from the NetScaler appliance.

Synopsis

```
rm lb vserver <name>@ ...
```

Arguments

name

Name of the virtual server.

Example

`rm vsrver lb_vip` To remove multiple vservers use the following command: `rm vsrver lb_vip`

set lb vsrver

Modifies the specified parameters of a load balancing virtual server.

Synopsys

```
set lb vsrver <name>@ [-IPAddress <ip_addr|ipv6_addr|*>@] [-IPPattern <ippat>] [-IPMask <ipmask>] [-weight
<positive_integer> <serviceName>@] [-persistenceType <persistenceType>] [-timeout <mins>] [-persistenceBackup
( SOURCEIP | NONE )] [-backupPersistenceTimeout <mins>] [-lbMethod <lbMethod>] [-hashLength
<positive_integer>] [-netmask <netmask>] [-v6netmasklen <positive_integer>] [-rule <expression>] [-cookieName
<string>] [-resRule <expression>] [-persistMask <netmask>] [-v6persistmasklen <positive_integer>] [-pq ( ON | OFF
)] [-sc ( ON | OFF )] [-rtspNat ( ON | OFF )] [-m <m>] [-tosld <positive_integer>] [-dataLength <positive_integer>] [-
dataOffset <positive_integer>] [-sessionless ( ENABLED | DISABLED )] [-connfailover <connfailover>] [-
backupVServer <string>] [-redirectURL <URL>] [-cacheable ( YES | NO )] [-cltTimeout <secs>] [-soMethod
<soMethod>] [-soThreshold <positive_integer>] [-soPersistence ( ENABLED | DISABLED )] [-soPersistenceTimeOut
<positive_integer>] [-healthThreshold <positive_integer>] [-soBackupAction <soBackupAction>] [-
redirectPortRewrite ( ENABLED | DISABLED )] [-downStateFlush ( ENABLED | DISABLED )] [-insertVserverIPPort
<insertVserverIPPort>] [-vipHeader <vipHeader>] [-disablePrimaryOnDown ( ENABLED | DISABLED )] [-AuthenticationHost
<string>] [-Authentication ( ON | OFF )] [-authn401 ( ON | OFF )] [-authnVsName <string>] [-push ( ENABLED |
DISABLED )] [-pushVserver <string>] [-pushLabel <expression>] [-pushMultiClients ( YES | NO )] [-Listenpolicy
<expression>] [-Listenpriority <positive_integer>] [-tcpProfileName <string>] [-httpProfileName <string>] [-
dbProfileName <string>] [-comment <string>] [-l2Conn ( ON | OFF )] [-oracleServerVersion ( 10G | 11G )] [-
mssqlServerVersion <mssqlServerVersion>] [-mysqlProtocolVersion <positive_integer>] [-mysqlServerVersion
<string>] [-mysqlCharacterSet <positive_integer>] [-mysqlServerCapabilities <positive_integer>] [-appflowLog (
ENABLED | DISABLED )] [-netProfile <string>] [-icmpVsrResponse ( PASSIVE | ACTIVE )] [-RHlstate ( PASSIVE |
ACTIVE )] [-newServiceRequest <positive_integer>] [-<newServiceRequestUnit>] [-
newServiceRequestIncrementInterval <positive_integer>] [-minAutoscaleMembers <positive_integer>] [-
maxAutoscaleMembers <positive_integer>] [-persistAVPno <positive_integer> ...] [-skippersistency
<skippersistency>] [-authnProfile <string>] [-macmodeRetainvlan ( ENABLED | DISABLED )] [-dbsLb ( ENABLED |
DISABLED )] [-dns64 ( ENABLED | DISABLED )] [-bypassAAAA ( YES | NO )] [-RecursionAvailable ( YES | NO )] [-
processLocal ( ENABLED | DISABLED )]
```

Arguments

name

Name of the virtual server.

IPAddress

IPv4 or IPv6 address to assign to the virtual server.

IPPattern

IP address pattern, in dotted decimal notation, for identifying packets to be accepted by the virtual server. The IP Mask parameter specifies which part of the destination IP address is matched against the pattern. Mutually exclusive with the IP Address parameter.

For example, if the IP pattern assigned to the virtual server is 198.51.100.0 and the IP mask is 255.255.240.0 (a forward mask), the first 20 bits in the destination IP addresses are matched with the first 20 bits in the pattern. The virtual server accepts requests with IP addresses that range from 198.51.96.1 to 198.51.111.254. You can also use a pattern such as 0.0.2.2 and a mask such as 0.0.255.255 (a reverse mask).

If a destination IP address matches more than one IP pattern, the pattern with the longest match is selected, and the associated virtual server processes the request. For example, if virtual servers vs1 and vs2 have the same IP pattern, 0.0.100.128, but different IP masks of 0.0.255.255 and 0.0.224.255, a destination IP address of 198.51.100.128 has the longest match with the IP pattern of vs1. If a destination IP address matches two or more virtual servers to the same extent, the request is processed by the virtual server whose port number matches the port number in the request.

IPMask

IP mask, in dotted decimal notation, for the IP Pattern parameter. Can have leading or trailing non-zero octets (for example, 255.255.240.0 or 0.0.255.255). Accordingly, the mask specifies whether the first n bits or the last n bits of the destination IP address in a client request are to be matched with the corresponding bits in the IP pattern. The former is called a forward mask. The latter is called a reverse mask.

weight

Weight to assign to the specified service.

Minimum value: 1

Maximum value: 100

serviceName

Service to bind to the virtual server.

persistenceType

Type of persistence for the virtual server. Available settings function as follows:

- * SOURCEIP - Connections from the same client IP address belong to the same persistence session.
- * COOKIEINSERT - Connections that have the same HTTP Cookie, inserted by a Set-Cookie directive from a server, belong to the same persistence session.
- * SSLSESSION - Connections that have the same SSL Session ID belong to the same persistence session.
- * CUSTOMSERVERID - Connections with the same server ID form part of the same session. For this persistence type, set the Server ID (CustomServerID) parameter for each service and configure the Rule parameter to identify the server ID in a request.
- * RULE - All connections that match a user defined rule belong to the same persistence session.
- * URLPASSIVE - Requests that have the same server ID in the URL query belong to the same persistence session. The server ID is the hexadecimal representation of the IP address and port of the service to which the request must be forwarded. This persistence type requires a rule to identify the server ID in the request.
- * DESTIP - Connections to the same destination IP address belong to the same persistence session.
- * SRCIPDESTIP - Connections that have the same source IP address and destination IP address belong to the same persistence session.
- * CALLID - Connections that have the same CALL-ID SIP header belong to the same persistence session.
- * RTSPSID - Connections that have the same RTSP Session ID belong to the same persistence session.

Possible values: SOURCEIP, COOKIEINSERT, SSLSESSION, RULE, URLPASSIVE, CUSTOMSERVERID, DESTIP, SRCIPDESTIP, CALLID, RTSPSID, DIAMETER, NONE

timeout

Time period for which a persistence session is in effect.

Default value: 2

Maximum value: 1440

persistenceBackup

Backup persistence type for the virtual server. Becomes operational if the primary persistence mechanism fails.

Possible values: SOURCEIP, NONE

backupPersistenceTimeout

Time period for which backup persistence is in effect.

Default value: 2

Minimum value: 2

Maximum value: 1440

lbMethod

Load balancing method. The available settings function as follows:

- * ROUNDROBIN - Distribute requests in rotation, regardless of the load. Weights can be assigned to services to enforce weighted round robin distribution.
- * LEASTCONNECTION (default) - Select the service with the fewest connections.
- * LEASTRESPONSETIME - Select the service with the lowest average response time.
- * LEASTBANDWIDTH - Select the service currently handling the least traffic.
- * LEASTPACKETS - Select the service currently serving the lowest number of packets per second.
- * CUSTOMLOAD - Base service selection on the SNMP metrics obtained by custom load monitors.
- * LRTM - Select the service with the lowest response time. Response times are learned through monitoring probes. This method also takes the number of active connections into account.

Also available are a number of hashing methods, in which the appliance extracts a predetermined portion of the request, creates a hash of the portion, and then checks whether any previous requests had the same hash value. If it finds a match, it forwards the request to the service that served those previous requests. Following are the hashing methods:

- * URLHASH - Create a hash of the request URL (or part of the URL).
- * DOMAINHASH - Create a hash of the domain name in the request (or part of the domain name). The domain name is taken from either the URL or the Host header. If the domain name appears in both locations, the URL is preferred. If the request does not contain a domain name, the load balancing method defaults to LEASTCONNECTION.
- * DESTINATIONIPHASH - Create a hash of the destination IP address in the IP header.
- * SOURCEIPHASH - Create a hash of the source IP address in the IP header.
- * TOKEN - Extract a token from the request, create a hash of the token, and then select the service to which any previous requests with the same token hash value were sent.
- * SRCIPDESTIPHASH - Create a hash of the string obtained by concatenating the source IP address and destination IP address in the IP header.
- * SRCIPSRCPORHASH - Create a hash of the source IP address and source port in the IP header.
- * CALLIDHASH - Create a hash of the SIP Call-ID header.

Possible values: ROUNDROBIN, LEASTCONNECTION, LEASTRESPONSETIME, URLHASH, DOMAINHASH, DESTINATIONIPHASH, SOURCEIPHASH, SRCIPDESTIPHASH, LEASTBANDWIDTH, LEASTPACKETS, TOKEN, SRCIPSRCPORHASH, LRTM, CALLIDHASH, CUSTOMLOAD, LEASTREQUEST

Default value: LEASTCONNECTION

hashLength

Number of bytes to consider for the hash value used in the URLHASH and DOMAINHASH load balancing methods.

Default value: 80

Minimum value: 1

Maximum value: 4096

netmask

IPv4 subnet mask to apply to the destination IP address or source IP address when the load balancing method is DESTINATIONIPHASH or SOURCEIPHASH.

Default value: 0xFFFFFFFF

v6netmasklen

Number of bits to consider in an IPv6 destination or source IP address, for creating the hash that is required by the DESTINATIONIPHASH and SOURCEIPHASH load balancing methods.

Default value: 128

Minimum value: 1

Maximum value: 128

rule

Expression, or name of a named expression, against which traffic is evaluated. Written in the classic or default syntax.

Note:

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

The following requirements apply only to the NetScaler CLI:

- * If the expression includes one or more spaces, enclose the entire expression in double quotation marks.
- * If the expression itself includes double quotation marks, escape the quotations by using the \\ character.
- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

Default value: "none"

cookieName

Use this parameter to specify the cookie name for COOKIE persistence type. It specifies the name of cookie with a maximum of 32 characters. If not specified, cookie name is internally generated.

resRule

Default syntax expression specifying which part of a server's response to use for creating rule based persistence sessions (persistence type RULE). Can be either an expression or the name of a named expression.

Example:

```
HTTP.RES.HEADER("setcookie").VALUE(0).TYPECAST_NVLIST_T('=';').VALUE("server1").
```

Default value: "none"

persistMask

Persistence mask for IP based persistence types, for IPv4 virtual servers.

Default value: 0xFFFFFFFF

v6persistmasklen

Persistence mask for IP based persistence types, for IPv6 virtual servers.

Default value: 128

Minimum value: 1

Maximum value: 128

pq

Use priority queuing on the virtual server. based persistence types, for IPv6 virtual servers.

Possible values: ON, OFF

Default value: OFF

sc

Use SureConnect on the virtual server.

Possible values: ON, OFF

Default value: OFF

rtspNat

Use network address translation (NAT) for RTSP data connections.

Possible values: ON, OFF

Default value: OFF

m

Redirection mode for load balancing. Available settings function as follows:

- * IP - Before forwarding a request to a server, change the destination IP address to the server's IP address.
- * MAC - Before forwarding a request to a server, change the destination MAC address to the server's MAC address. The destination IP address is not changed. MAC-based redirection mode is used mostly in firewall load balancing deployments.
- * IPTUNNEL - Perform IP-in-IP encapsulation for client IP packets. In the outer IP headers, set the destination IP address to the IP address of the server and the source IP address to the subnet IP (SNIP). The client IP packets are not modified. Applicable to both IPv4 and IPv6 packets.
- * TOS - Encode the virtual server's TOS ID in the TOS field of the IP header.

You can use either the IPTUNNEL or the TOS option to implement Direct Server Return (DSR).

Possible values: IP, MAC, IPTUNNEL, TOS

Default value: IP

tosId

TOS ID of the virtual server. Applicable only when the load balancing redirection mode is set to TOS.

Minimum value: 1

Maximum value: 63

dataLength

Length of the token to be extracted from the data segment of an incoming packet, for use in the token method of load balancing. The length of the token, specified in bytes, must not be greater than 24 KB. Applicable to virtual servers of type TCP.

Minimum value: 1

Maximum value: 100

dataOffset

Offset to be considered when extracting a token from the TCP payload. Applicable to virtual servers, of type TCP, using the token method of load balancing. Must be within the first 24 KB of the TCP payload.

Minimum value: 0

Maximum value: 25400

sessionless

Perform load balancing on a per-packet basis, without establishing sessions. Recommended for load balancing of intrusion detection system (IDS) servers and scenarios involving direct server return (DSR), where session information is unnecessary.

Possible values: ENABLED, DISABLED

Default value: DISABLED

connfailover

Mode in which the connection failover feature must operate for the virtual server. After a failover, established TCP connections and UDP packet flows are kept active and resumed on the secondary appliance. Clients remain connected to the same servers. Available settings function as follows:

* STATEFUL - The primary appliance shares state information with the secondary appliance, in real time, resulting in some runtime processing overhead.

* STATELESS - State information is not shared, and the new primary appliance tries to re-create the packet flow on the basis of the information contained in the packets it receives.

* DISABLED - Connection failover does not occur.

Possible values: DISABLED, STATEFUL, STATELESS

Default value: DISABLED

backupVServer

Name of the backup virtual server to which to forward requests if the primary virtual server goes DOWN or reaches its spillover threshold.

redirectURL

URL to which to redirect traffic if the virtual server becomes unavailable.

WARNING! Make sure that the domain in the URL does not match the domain specified for a content switching policy. If it does, requests are continuously redirected to the unavailable virtual server.

cacheable

Route cacheable requests to a cache redirection virtual server. The load balancing virtual server can forward requests only to a transparent cache redirection virtual server that has an IP address and port combination of *:80, so such a cache redirection virtual server must be configured on the appliance.

Possible values: YES, NO

Default value: NO

cltTimeout

Idle time, in seconds, after which a client connection is terminated.

Default value: -1

Maximum value: 31536000

soMethod

Type of threshold that, when exceeded, triggers spillover. Available settings function as follows:

* CONNECTION - Spillover occurs when the number of client connections exceeds the threshold.

* DYNAMICCONNECTION - Spillover occurs when the number of client connections at the virtual server exceeds the sum of the maximum client (Max Clients) settings for bound services. Do not specify a spillover threshold for this setting, because the threshold is implied by the Max Clients settings of bound services.

* **BANDWIDTH** - Spillover occurs when the bandwidth consumed by the virtual server's incoming and outgoing traffic exceeds the threshold.

* **HEALTH** - Spillover occurs when the percentage of weights of the services that are UP drops below the threshold. For example, if services svc1, svc2, and svc3 are bound to a virtual server, with weights 1, 2, and 3, and the spillover threshold is 50%, spillover occurs if svc1 and svc3 or svc2 and svc3 transition to DOWN.

* **NONE** - Spillover does not occur.

Possible values: CONNECTION, DYNAMICCONNECTION, BANDWIDTH, HEALTH, NONE

soThreshold

Threshold at which spillover occurs. Specify an integer for the CONNECTION spillover method, a bandwidth value in kilobits per second for the BANDWIDTH method (do not enter the units), or a percentage for the HEALTH method (do not enter the percentage symbol).

Minimum value: 1

Maximum value: 4294967287

soPersistence

If spillover occurs, maintain source IP address based persistence for both primary and backup virtual servers.

Possible values: ENABLED, DISABLED

Default value: DISABLED

soPersistenceTimeOut

Timeout for spillover persistence, in minutes.

Default value: 2

Minimum value: 2

Maximum value: 1440

healthThreshold

Threshold in percent of active services below which vserver state is made down. If this threshold is 0, vserver state will be up even if one bound service is up.

Default value: 0

Minimum value: 0

Maximum value: 100

soBackupAction

Action to be performed if spillover is to take effect, but no backup chain to spillover is usable or exists

Possible values: DROP, ACCEPT, REDIRECT

redirectPortRewrite

Rewrite the port and change the protocol to ensure successful HTTP redirects from services.

Possible values: ENABLED, DISABLED

Default value: DISABLED

downStateFlush

Flush all active transactions associated with a virtual server whose state transitions from UP to DOWN. Do not enable this option for applications that must complete their transactions.

Possible values: ENABLED, DISABLED

Default value: ENABLED

insertVserverIPPort

Insert an HTTP header, whose value is the IP address and port number of the virtual server, before forwarding a request to the server. The format of the header is <vipHeader>: <virtual server IP address>_<port number>, where vipHeader is the name that you specify for the header. If the virtual server has an IPv6 address, the address in the header is enclosed in brackets ([and]) to separate it from the port number. If you have mapped an IPv4 address to a virtual server's IPv6 address, the value of this parameter determines which IP address is inserted in the header, as follows:

* VIPADDR - Insert the IP address of the virtual server in the HTTP header regardless of whether the virtual server has an IPv4 address or an IPv6 address. A mapped IPv4 address, if configured, is ignored.

* V6TOV4MAPPING - Insert the IPv4 address that is mapped to the virtual server's IPv6 address. If a mapped IPv4 address is not configured, insert the IPv6 address.

* OFF - Disable header insertion.

Possible values: OFF, VIPADDR, V6TOV4MAPPING

vipHeader

Name for the inserted header. The default name is vip-header.

disablePrimaryOnDown

If the primary virtual server goes down, do not allow it to return to primary status until manually enabled.

Possible values: ENABLED, DISABLED

Default value: DISABLED

AuthenticationHost

Fully qualified domain name (FQDN) of the authentication virtual server to which the user must be redirected for authentication. Make sure that the Authentication parameter is set to ENABLED.

Authentication

Enable or disable user authentication.

Possible values: ON, OFF

Default value: OFF

authn401

Enable or disable user authentication with HTTP 401 responses.

Possible values: ON, OFF

Default value: OFF

authnVsName

Name of an authentication virtual server with which to authenticate users.

push

Process traffic with the push virtual server that is bound to this load balancing virtual server.

Possible values: ENABLED, DISABLED

Default value: DISABLED

pushVserver

Name of the load balancing virtual server, of type PUSH or SSL_PUSH, to which the server pushes updates received on the load balancing virtual server that you are configuring.

pushLabel

Expression for extracting a label from the server's response. Can be either an expression or the name of a named expression.

Default value: "none"

pushMultiClients

Allow multiple Web 2.0 connections from the same client to connect to the virtual server and expect updates.

Possible values: YES, NO

Default value: NO

Listenpolicy

Default syntax expression identifying traffic accepted by the virtual server. Can be either an expression (for example, CLIENT.IP.DST.IN_SUBNET(192.0.2.0/24) or the name of a named expression. In the above example, the virtual server accepts all requests whose destination IP address is in the 192.0.2.0/24 subnet.

Default value: "none"

Listenpriority

Integer specifying the priority of the listen policy. A higher number specifies a lower priority. If a request matches the listen policies of more than one virtual server the virtual server whose listen policy has the highest priority (the lowest priority number) accepts the request.

Default value: 101

Minimum value: 0

Maximum value: 101

tcpProfileName

Name of the TCP profile whose settings are to be applied to the virtual server.

httpProfileName

Name of the HTTP profile whose settings are to be applied to the virtual server.

dbProfileName

Name of the DB profile whose settings are to be applied to the virtual server.

comment

Any comments that you might want to associate with the virtual server.

l2Conn

Use Layer 2 parameters (channel number, MAC address, and VLAN ID) in addition to the 4-tuple (<source IP>:<source port>::<destination IP>:<destination port>) that is used to identify a connection. Allows multiple TCP and non-TCP connections with the same 4-tuple to co-exist on the NetScaler appliance.

Possible values: ON, OFF

oracleServerVersion

Oracle server version

Possible values: 10G, 11G

Default value: 10G

mssqlServerVersion

For a load balancing virtual server of type MSSQL, the Microsoft SQL Server version. Set this parameter if you expect some clients to run a version different from the version of the database. This setting provides

compatibility between the client-side and server-side connections by ensuring that all communication conforms to the server's version.

Possible values: 70, 2000, 2000SP1, 2005, 2008, 2008R2, 2012

Default value: 2008R2

mysqlProtocolVersion

MySQL protocol version that the virtual server advertises to clients.

Default value: NSA_MYSQL_PROTOCOL_VER_DEFAULT

Minimum value: 0

mysqlServerVersion

MySQL server version string that the virtual server advertises to clients.

Default value: NSA_MYSQL_SERVER_VER_DEFAULT

mysqlCharacterSet

Character set that the virtual server advertises to clients.

Default value: NSA_MYSQL_CHAR_SET_DEFAULT

Minimum value: 0

mysqlServerCapabilities

Server capabilities that the virtual server advertises to clients.

Default value: NSA_MYSQL_SVR_CAPABILITIES_DEFAULT

Minimum value: 0

appflowLog

Apply AppFlow logging to the virtual server.

Possible values: ENABLED, DISABLED

Default value: ENABLED

netProfile

Name of the network profile to associate with the virtual server. If you set this parameter, the virtual server uses only the IP addresses in the network profile as source IP addresses when initiating connections with servers.

icmpVsrResponse

How the NetScaler appliance responds to ping requests received for an IP address that is common to one or more virtual servers. Available settings function as follows:

* If set to PASSIVE on all the virtual servers that share the IP address, the appliance always responds to the ping requests.

* If set to ACTIVE on all the virtual servers that share the IP address, the appliance responds to the ping requests if at least one of the virtual servers is UP. Otherwise, the appliance does not respond.

* If set to ACTIVE on some virtual servers and PASSIVE on the others, the appliance responds if at least one virtual server with the ACTIVE setting is UP. Otherwise, the appliance does not respond.

Note: This parameter is available at the virtual server level. A similar parameter, ICMP Response, is available at the IP address level, for IPv4 addresses of type VIP. To set that parameter, use the add ip command in the CLI or the Create IP dialog box in the GUI.

Possible values: PASSIVE, ACTIVE

Default value: PASSIVE

RHIstate

Route Health Injection (RHI) functionality of the NetScaler appliance for advertising the route of the VIP address associated with the virtual server. When Vserver RHI Level (RHI) parameter is set to VSVR_CNTRLD, the following are different RHI behaviors for the VIP address on the basis of RHIstate (RHI STATE) settings on the virtual servers associated with the VIP address:

* If you set RHI STATE to PASSIVE on all virtual servers, the NetScaler ADC always advertises the route for the VIP address.

* If you set RHI STATE to ACTIVE on all virtual servers, the NetScaler ADC advertises the route for the VIP address if at least one of the associated virtual servers is in UP state.

* If you set RHI STATE to ACTIVE on some and PASSIVE on others, the NetScaler ADC advertises the route for the VIP address if at least one of the associated virtual servers, whose RHI STATE set to ACTIVE, is in UP state.

Possible values: PASSIVE, ACTIVE

Default value: PASSIVE

newServiceRequest

Number of requests, or percentage of the load on existing services, by which to increase the load on a new service at each interval in slow-start mode. A non-zero value indicates that slow-start is applicable. A zero value indicates that the global RR startup parameter is applied. Changing the value to zero will cause services currently in slow start to take the full traffic as determined by the LB method. Subsequently, any new services added will use the global RR factor.

Default value: 0

Minimum value: 0

newServiceRequestUnit

Units in which to increment load at each interval in slow-start mode.

Possible values: PER_SECOND, PERCENT

Default value: PER_SECOND

newServiceRequestIncrementInterval

Interval, in seconds, between successive increments in the load on a new service or a service whose state has just changed from DOWN to UP. A value of 0 (zero) specifies manual slow start.

Default value: 0

Minimum value: 0

Maximum value: 3600

minAutoscaleMembers

Minimum number of members expected to be present when vserver is used in Autoscale.

Default value: 0

Minimum value: 0

Maximum value: 5000

maxAutoscaleMembers

Maximum number of members expected to be present when vserver is used in Autoscale.

Default value: 0

Minimum value: 0

Maximum value: 5000

persistAVPno

Persist AVP number for Diameter Persistency.

In case this AVP is not defined in Base RFC 3588 and it is nested inside a Grouped AVP, define a sequence of AVP numbers (max 3) in order of parent to child. So say persist AVP number X is nested inside AVP Y which is nested in Z, then define the list as Z Y X

Minimum value: 1

skippersistency

This argument decides the behavior incase the service which is selected from an existing persistence session has reached threshold.

Possible values: Bypass, ReLb, None

Default value: None

authnProfile

Name of the authentication profile to be used when authentication is turned on.

macmodeRetainvlan

This option is used to retain vlan information of incoming packet when macmode is enabled

Possible values: ENABLED, DISABLED

Default value: DISABLED

dbslb

Enable database specific load balancing for MySQL and MSSQL service types.

Possible values: ENABLED, DISABLED

Default value: DISABLED

dns64

This argument is for enabling/disabling the dns64 on lbvserver

Possible values: ENABLED, DISABLED

bypassAAAA

If this option is enabled while resolving DNS64 query AAAA queries are not sent to back end dns server

Possible values: YES, NO

Default value: NO

RecursionAvailable

When set to YES, this option causes the DNS replies from this vserver to have the RA bit turned on. Typically one would set this option to YES, when the vserver is load balancing a set of DNS servers that support recursive queries.

Possible values: YES, NO

Default value: NO

processLocal

By turning on this option packets destined to a vserver in a cluster will not under go any steering. Turn this option for single packet request response mode or when the upstream device is performing a proper RSS for connection based distribution.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
set lb vserver http_vip -lbmethod LEASTRESPONSETIME To set the load balancing method for
```

unset lb vserver

Removes the specified parameter settings from the virtual server..Refer to the set lb vserver command for meanings of the arguments.

Synopsis

```
unset lb vserver <name>@ [-backupVServer] [-cltTimeout] [-redirectURL] [-authn401] [-Authentication] [-
AuthenticationHost] [-authnVsName] [-pushVserver] [-pushLabel] [-tcpProfileName] [-httpProfileName] [-
dbProfileName] [-rule] [-l2Conn] [-mysqlProtocolVersion] [-mysqlServerVersion] [-mysqlCharacterSet] [-
mysqlServerCapabilities] [-appflowLog] [-netProfile] [-icmpVsrResponse] [-skippersistency] [-minAutoscaleMembers]
[-maxAutoscaleMembers] [-authnProfile] [-macmodeRetainvlan] [-dbsLb] [-serviceName] [-persistenceType] [-
timeout] [-persistenceBackup] [-backupPersistenceTimeout] [-lbMethod] [-hashLength] [-netmask] [-v6netmasklen] [-
cookieName] [-resRule] [-persistMask] [-v6persistmasklen] [-pq] [-sc] [-rtspNat] [-m] [-tosId] [-dataLength] [-
dataOffset] [-sessionless] [-connfailover] [-cacheable] [-soMethod] [-soPersistence] [-soPersistenceTimeOut] [-
healthThreshold] [-soBackupAction] [-redirectPortRewrite] [-downStateFlush] [-insertVserverIPPort] [-vipHeader] [-
disablePrimaryOnDown] [-push] [-pushMultiClients] [-Listenpolicy] [-Listenpriority] [-comment] [-oracleServerVersion]
[-mssqlServerVersion] [-RHlstate] [-newServiceRequest] [-newServiceRequestUnit] [-
newServiceRequestIncrementInterval] [-persistAVPno] [-RecursionAvailable]
```

Example

```
unset lb vserver lb_vip -backupVServer To unset the backup virtual server for multiple v:
```

bind lb vserver

Binds a service, service group, or policy to a virtual server.

Synopsis

```
bind lb vserver <name>@ ((<serviceName>@ [-weight <positive_integer>] ) | <serviceGroupName>@ | (-
policyName <string>@ [-priority <positive_integer>] [-gotoPriorityExpression <expression>] [-type ( REQUEST |
RESPONSE )] [-invoke (<labelType> <labelName> ) ] ))
```

Arguments

name

Name for the virtual server. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at sign (@), equal sign (=), and hyphen (-) characters. Can be changed after the virtual server is created.

CLI Users: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my vserver" or 'my vserver').

serviceName

Name of the service.

weight

Integer specifying the weight of the service. A larger number specifies a greater weight. Defines the capacity of the service relative to the other services in the load balancing configuration. Determines the priority given to the service in load balancing decisions.

Default value: 1

Minimum value: 1

Maximum value: 100

serviceGroupName

Name of the service group.

policyName

Name of the policy to bind to the virtual server.

priority

Integer specifying the policy's priority. The lower the priority number, the higher the policy's priority.

Minimum value: 1

Maximum value: 2147483647

gotoPriorityExpression

Expression or other value specifying the next policy to be evaluated if the current policy evaluates to TRUE. Specify one of the following values:

- * NEXT - Evaluate the policy with the next higher priority number.
- * END - End policy evaluation.
- * USE_INVOCATION_RESULT - Applicable if this policy invokes another policy label. If the final goto in the invoked policy label has a value of END, the evaluation stops. If the final goto is anything other than END, the current policy label performs a NEXT.
- * A default syntax expression that evaluates to a number.

If you specify an expression, the number to which it evaluates determines the next policy to evaluate, as follows:

- * If the expression evaluates to a higher numbered priority, the policy with that priority is evaluated next.
- * If the expression evaluates to the priority of the current policy, the policy with the next higher numbered priority is evaluated next.
- * If the expression evaluates to a priority number that is numerically higher than the highest numbered priority, policy evaluation ends.

An UNDEF event is triggered if:

- * The expression is invalid.
- * The expression evaluates to a priority number that is numerically lower than the current policy's priority.
- * The expression evaluates to a priority number that is between the current policy's priority number (say, 30) and the highest priority number (say, 100), but does not match any configured priority number (for example, the expression evaluates to the number 85). This example assumes that the priority number increments by 10 for every successive policy, and therefore a priority number of 85 does not exist in the policy label.

type

Bind point to which to bind the policy. Applicable only to compression, rewrite, and cache policies.

Possible values: REQUEST, RESPONSE

invoke

Invoke policies bound to a virtual server or policy label.

labelType

Type of policy label to invoke. Applicable only to rewrite and cache policies. Available settings function as follows:

- * reqvserver - Evaluate the request against the request-based policies bound to the specified virtual server.
- * resvserver - Evaluate the response against the response-based policies bound to the specified virtual server.
- * policylabel - invoke the request or response against the specified user-defined policy label.

Possible values: reqvserver, resvserver, policylabel

labelName

Name of the virtual server or user-defined policy label to invoke if the policy evaluates to TRUE.

Example

```
bind lb vserver http_vip http_svc To bind a service to multiple vservers use the followin
```

unbind lb vserver

Unbinds a service, service group, or policy from a virtual server.

Synopsys

```
unbind lb vserver <name>@ (<serviceName>@ | <serviceGroupName>@ | (-policyName <string>@ [-type (REQUEST | RESPONSE )])) [-priority <positive_integer>]
```

Arguments

name

Name for the virtual server. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at sign (@), equal sign (=), and hyphen (-) characters. Can be changed after the virtual server is created.

CLI Users: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my vserver" or 'my vserver').

serviceName

Name of the service.

serviceGroupName

The name of the service group that is unbound.

policyName

Name of the policy to bind to the virtual server.

type

Bind point from which to unbind the policy label.

Possible values: REQUEST, RESPONSE

priority

Priority number of the policy.

Minimum value: 1

Maximum value: 2147483647

Example

`unbind lb vserver http_vip http_svc` To unbind a service from multiple vservers use the :

enable lb vserver

Enables a virtual server.

Synopsys

`enable lb vserver <name>@`

Arguments

name

Name of the virtual server.

Example

`enable vserver lb_vip` To enable multiple vservers at once use the following command: `en`

disable lb vserver

Disables a virtual server.

Synopsys

`disable lb vserver <name>@`

Arguments

name

Name of the virtual server.

Example

`disable vserver lb_vip` To disable multiple vservers at once use the following command: `d`

show lb vserver

Displays the statistical data collected for a load balancing virtual server.

Synopsys

`show lb vserver [<name>]` `show lb vserver stats` - alias for 'stat lb vserver'

Arguments

name

Name of the virtual server. If no name is provided, statistical data of all configured virtual servers is displayed.

Outputs

insertVserverIPPort

The virtual IP and port header insertion option for the vserver.

vipHeader

Name for the inserted header. The default name is vip-header.

value

SSL status.

stateflag**appfwPolicyFlag****IPAddress**

IPv4 or IPv6 address to assign to the virtual server.

IPPattern

The IP pattern of the virtual server.

IPMask

The IP address mask of the virtual server.

Listenpolicy

The string is listenpolicy configured for lb vserver

Listenpriority

This parameter is the priority for listen policy of LB Vserver.

IPMapping

The permanent mapping for the V6 Address

port

Port number for the virtual server.

range

Number of IP addresses that the appliance must generate and assign to the virtual server. The virtual server then functions as a network virtual server, accepting traffic on any of the generated IP addresses. The IP addresses are generated automatically, as follows:

* For a range of n, the last octet of the address specified by the IP Address parameter increments n-1 times.

* If the last octet exceeds 255, it rolls over to 0 and the third octet increments by 1.

Note: The Range parameter assigns multiple IP addresses to one virtual server. To generate an array of virtual servers, each of which owns only one IP address, use brackets in the IP Address and Name parameters to specify the range. For example:

```
add lb vserver my_vserver[1-3] HTTP 192.0.2.[1-3] 80
```

serviceType

Protocol used by the service (also called the service type).

ngname

Nodegroup name to which this lbvserver belongs to

type

The bindpoint to which the policy is bound

state

State of the load balancing virtual server.

effectiveState

Effective state of the LB vserver , based on the state of backup vservers.

status

Current status of the lb vserver. During the initial phase if the configured lb method is not round robin , the vserver will adopt round robin to distribute traffic for a predefined number of requests.

lbrreason

Reason why a vserver is in RR. The following are the reasons:

- 1 - MEP is DOWN (GSLB)
- 2 - LB method has changed
- 3 - Bound service's state changed to UP
- 4 - A new service is bound
- 5 - Startup RR factor has changed
- 6 - LB feature is enabled
- 7 - Load monitor is not active on a service
- 8 - Vserver is Enabled
- 9 - SSL feature is Enabled
- 10 - All bound services have reached threshold. Using effective state to load balance (GSLB)
- 11 - Primary state of bound services are not UP. Using effective state to load balance (GSLB)
- 12 - No LB decision can be made as all bound services have either reached threshold or are not UP (GSLB)
- 13 - All load monitors are active

cacheType

Cache type.

redirect

Cache redirect type.

precedence

Precedence.

redirectURL

The redirect URL.

Authentication

Authentication.

authn401

HTTP 401 response based authentication.

authnVsName

Name of an authentication virtual server with which to authenticate users.

homePage

Home page.

dnsVserverName

DNS vserver name.

domain

Domain.

policyName

Name of the policy bound to the LB vserver.

serviceName

Service to bind to the virtual server.

serviceGroupName

The service group name bound to the selected load balancing virtual server.

weight

Weight to assign to the specified service.

dynamicWeight

Dynamic weight

cacheVserver

Cache virtual server.

backupVServer

Name of the backup virtual server to which to forward requests if the primary virtual server goes DOWN or reaches its spillover threshold.

priority

Priority.

cltTimeout

The client timeout in seconds.

soMethod

The spillover method to be in effect.

soPersistence

State of spillover persistence.

soPersistenceTimeOut

The maximum time persistence is in effect for a specific client on a spillover vserver.

healthThreshold

Threshold in percent of active services below which vserver state is made down.

soThreshold

Threshold at which spillover occurs. Specify an integer for the CONNECTION spillover method, a bandwidth value in kilobits per second for the BANDWIDTH method (do not enter the units), or a percentage for the HEALTH method (do not enter the percentage symbol).

soBackupAction

Action to be performed if spillover is to take effect, but no backup chain to spillover is usable or exists

lbMethod

The load balancing method to be in effect

hashLength

The hash length.

dataOffset

The data offset length for TOKEN load balancing method.

health

Health of vserver based on percentage of weights of active svcs/all svcs. This does not consider administratively disabled svcs

dataLength

The data length for TOKEN load balancing method.

netmask

The netmask of the destination network.

v6netmasklen

The netmask of the destination network.

rule

Rule type.

resRule

Use this parameter to specify the expression to be used in response for RULE persistence type.

The string is an in-line expression with a maximum of 1499 characters.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

ruleType

Rule type.

groupName

LB group to which the lb vserver is to be bound.

m

The LB mode.

tosId

TOS ID

persistenceType

The persistence type for the specified virtual server

timeout

The maximum time persistence is in effect for a specific client.

cookieDomain

Domain name to be used in the set cookie header in case of cookie persistence.

persistMask

The persistence mask for v4 traffic

v6persistmasklen

The persistence mask for v6 traffic.

persistenceBackup

The maximum time backup persistence is in effect for a specific client.

backupPersistenceTimeout

Time period for which backup persistence is in effect.

cacheable

The state of caching.

pq

The state of priority queuing on the specified virtual server.

sc

The state of SureConnect the specified virtual server.

rtspNat

Use network address translation (NAT) for RTSP data connections.

sessionless

To enable sessionless load balancing, enable this option

map

Map.

connfailover

The connection failover mode of the virtual server

redirectPortRewrite

Rewrite the port and change the protocol to ensure successful HTTP redirects from services.

downStateFlush

Flush all active transactions associated with a virtual server whose state transitions from UP to DOWN. Do not enable this option for applications that must complete their transactions.

disablePrimaryOnDown

Tells whether traffic will continue reaching backup vservers even after primary comes UP from DOWN state.

gt2GB

Allow for greater than 2 GB transactions on this vserver.

consolidatedLConn

Use consolidated stats for LeastConnection.

consolidatedLConnGbl

Fetches Global setting.

thresholdValue

Tells whether threshold exceeded for this service participating in CUSTOMLB

invoke

Invoke policies bound to a virtual server or policy label.

labelType

The invocation type.

labelName

Name of the label invoked.

cookieIpPort

Encrypted Ip address and port of the service that is inserted into the set-cookie http header

cookieName

Use this parameter to specify the cookie name for COOKIE persistence type. It specifies the name of cookie with a maximum of 32 characters. If not specified, cookie name is internally generated.

vserverId

Vserver Id

version

Cookie version

totalServices

Total number of services bound to the vserver.

activeServices

Total number of active services bound to the vserver.

stateChangeTimeSec

Time when last state change happened. Seconds part.

stateChangeTimeSeconds

Time when last state change happened. Seconds part.

stateChangeTimeSec

Time at which last state change happened. Milliseconds part.

ticksSinceLastStateChange

Time in 10 millisecond ticks since the last state change.

hits

Number of hits.

piPolicyhits

Number of hits.

AuthenticationHost

Fully qualified domain name (FQDN) of the authentication virtual server to which the user must be redirected for authentication. Make sure that the Authentication parameter is set to ENABLED.

push

Process traffic with the push virtual server that is bound to this load balancing virtual server.

pushVserver

Name of the load balancing virtual server, of type PUSH or SSL_PUSH, to which the server pushes updates received on the load balancing virtual server that you are configuring.

pushLabel

Expression for extracting a label from the server's response. Can be either an expression or the name of a named expression.

pushMultiClients

Allow multiple Web 2.0 connections from the same client to connect to the virtual server and expect updates.

tcpProfileName

Name of the TCP profile whose settings are to be applied to the virtual server.

httpProfileName

Name of the HTTP profile whose settings are to be applied to the virtual server.

dbProfileName

Name of the DB profile whose settings are to be applied to the virtual server.

comment

Any comments that you might want to associate with the virtual server.

flag**flags****policySubType****l2Conn**

Use Layer 2 parameters (channel number, MAC address, and VLAN ID) in addition to the 4-tuple (<source IP>:<source port>:<destination IP>:<destination port>) that is used to identify a connection. Allows multiple TCP and non-TCP connections with the same 4-tuple to co-exist on the NetScaler appliance.

oracleServerVersion

Oracle server version

mssqlServerVersion

For a load balancing virtual server of type MSSQL, the Microsoft SQL Server version. Set this parameter if you expect some clients to run a version different from the version of the database. This setting provides compatibility between the client-side and server-side connections by ensuring that all communication conforms to the server's version.

mysqlProtocolVersion

MySQL protocol version that the virtual server advertises to clients.

mysqlServerVersion

MySQL server version string that the virtual server advertises to clients.

mysqlCharacterSet

Character set that the virtual server advertises to clients.

mysqlServerCapabilities

Server capabilities that the virtual server advertises to clients.

appflowLog

Apply AppFlow logging to the virtual server.

netProfile

Name of the network profile to associate with the virtual server. If you set this parameter, the virtual server uses only the IP addresses in the network profile as source IP addresses when initiating connections with servers.

isGslb

This field is set to true if it is a GSLB server.

icmpVsrResponse

How the NetScaler appliance responds to ping requests received for an IP address that is common to one or more virtual servers. Available settings function as follows:

- * If set to PASSIVE on all the virtual servers that share the IP address, the appliance always responds to the ping requests.
- * If set to ACTIVE on all the virtual servers that share the IP address, the appliance responds to the ping requests if at least one of the virtual servers is UP. Otherwise, the appliance does not respond.
- * If set to ACTIVE on some virtual servers and PASSIVE on the others, the appliance responds if at least one virtual server with the ACTIVE setting is UP. Otherwise, the appliance does not respond.

Note: This parameter is available at the virtual server level. A similar parameter, ICMP Response, is available at the IP address level, for IPv4 addresses of type VIP. To set that parameter, use the add ip command in the CLI or the Create IP dialog box in the GUI.

RHIstate

Route Health Injection (RHI) functionality of the NetScaler appliance for advertising the route of the VIP address associated with the virtual server. When Vserver RHI Level (RHI) parameter is set to VSVR_CNTRLD, the following are different RHI behaviors for the VIP address on the basis of RHIstate (RHI STATE) settings on the virtual servers associated with the VIP address:

- * If you set RHI STATE to PASSIVE on all virtual servers, the NetScaler ADC always advertises the route for the VIP address.
- * If you set RHI STATE to ACTIVE on all virtual servers, the NetScaler ADC advertises the route for the VIP address if at least one of the associated virtual servers is in UP state.
- * If you set RHI STATE to ACTIVE on some and PASSIVE on others, the NetScaler ADC advertises the route for the VIP address if at least one of the associated virtual servers, whose RHI STATE set to ACTIVE, is in UP state.

newServiceRequest

Number of requests, or percentage of the load on existing services, by which to increase the load on a new service at each interval in slow-start mode. A non-zero value indicates that slow-start is applicable. A zero value indicates that the global RR startup parameter is applied. Changing the value to zero will cause services currently in slow start to take the full traffic as determined by the LB method. Subsequently, any new services added will use the global RR factor.

newServiceRequestUnit

Units in which to increment load at each interval in slow-start mode.

newServiceRequestIncrementInterval

Interval, in seconds, between successive increments in the load on a new service or a service whose state has just changed from DOWN to UP. A value of 0 (zero) specifies manual slow start.

vsvrcfgflags

Contains the config info of vserver to be used at validation

vsvrbindsvcip

used for showing the ip of bound entities

vsvrbindsvcport

used for showing ports of bound entities

persistAVPno

Persist AVP number for Diameter Persistency.

In case this AVP is not defined in Base RFC 3588 and it is nested inside a Grouped AVP, define a sequence of AVP numbers (max 3) in order of parent to child. So say persist AVP number X is nested inside AVP Y which is nested in Z, then define the list as Z Y X

skippersistency

This argument decides the behavior incase the service which is selected from an existing persistence session has reached threshold.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

minAutoscaleMembers

Minimum number of members expected to be present when vserver is used in Autoscale.

maxAutoscaleMembers

Maximum number of members expected to be present when vserver is used in Autoscale.

authnProfile

Name of the authentication profile to be used when authentication is turned on.

macmodeRetainvlan

This option is used to retain vlan information of incoming packet when macmode is enabled

dbslb

Enable database specific load balancing for MySQL and MSSQL service types.

dns64

This argument is for enabling/disabling the dns64 on lbserver

bypassAAAA

If this option is enabled while resolving DNS64 query AAAA queries are not sent to back end dns server

RecursionAvailable

When set to YES, this option causes the DNS replies from this vserver to have the RA bit turned on. Typically one would set this option to YES, when the vserver is load balancing a set of DNS servers that support recursive queries.

processLocal

By turning on this option packets destined to a vserver in a cluster will not under go any steering. Turn this option for single packet request response mode or when the upstream device is performing a proper RSS for connection based distribution.

vsvrdynconnsothreshold

Spillover threshold for dynamic connection

devno**count**

stat lb vserver

Displays the statistical data collected for a load balancing virtual server.

Synopsys

```
stat lb vserver [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)] [-sortBy Hits [<sortOrder>]]
```

Arguments

name

Name of the virtual server. If no name is provided, statistical data of all configured virtual servers is displayed.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

sortBy

use this argument to sort by specific key

Possible values: Hits

sortOrder

use this argument to specify sort order

Possible values: ascending, descending

Default value: SORT_DESCENDING

Outputs

count

devno

stateflag

Outputs

s surgeQ (vSurgeQ)

Number of requests waiting on this vserver.

Current Client Est connections (CIntEstConn)

Number of client connections in ESTABLISHED state.

total INACTIVE services (inactSvcs)

number of INACTIVE services bound to a vserver

Vserver Health (Health)

Health of the vserver. This gives percentage of UP services bound to this vserver.

Vserver IP address (vsvrIP)

IP address of the vserver

Port (port)

The port on which the service is running.

Vserver protocol (Protocol)

Protocol associated with the vserver

State

Current state of the server. Possible values are UP, DOWN, UNKNOWN, OFS(Out of Service), TROFS (Transition Out of Service), TROFS_DOWN(Down When going Out of Service)

total ACTIVE services (actSvcs)

number of ACTIVE services bound to a vserver

Vserver hits (Hits)

Total vserver hits

Requests (Req)

Total number of requests received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

Responses (Rsp)

Number of responses received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

Request bytes (Reqb)

Total number of request bytes received on this service or virtual server.

Response bytes (Rspb)

Number of response bytes received by this service or virtual server.

Total Packets rcvd (PktRx)

Total number of packets received by this service or virtual server.

Total Packets sent (PktTx)

Total number of packets sent.

Current client connections (CIntConn)

Number of current client connections.

Current server connections (SvrConn)

Number of current connections to the actual servers behind the virtual server.

Requests in surge queue (SurgeQ)

Number of requests in the surge queue.

s surgeQs (SvcSurgeQ)

Total number of requests in the surge queues of all the services bound to this LB-vserver.

Spill Over Threshold (SOTresh)

Spill Over Threshold set on the VServer.

Spill Over Hits (NumSo)

Number of times vserver experienced spill over.

Labeled Connection (LblConn)

Number of Labeled connection on this vserver

Push Labeled Connection (PushLbl)

Number of labels for this push vserver.

Deferred Request (DefReq)

Number of deferred request on this vserver

Invalid Request/Response (IvldReqRsp)

Number invalid requests/responses on this vserver

Invalid Request/Response Dropped (IvldReqRspDrp)

Number invalid requests/responses dropped on this vserver

Current Server Est connections (SvrEstConn)

Number of server connections in ESTABLISHED state.

rename lb vserver

Renames a load balancing virtual server.

Synopsys

```
rename lb vserver <name>@ <newName>@
```

Arguments

name

Existing name of the virtual server.

newName

New name for the virtual server.

Example

```
rename lb vserver http_vsvr http_vsvr_new
```

lb wlm

The following operations can be performed on "lb wlm":

add | **rm** | **set** | **unset** | **show** | **bind** | **unbind**

add lb wlm

Add a Work Load Manager. NOTE: This command is deprecated. WLM feature has been deprecated from Kos onwards as classic build is not supported.

Synopsys

Arguments

wlmName

The name of the Work Load Manager.

IPAddress

The IP address of the WLM.

port

The port of the WLM.

LBUID

The LBUID for the Load Balancer to communicate to the Work Load Manager.

KATimeout

The idle time period after which NS would probe the WLM. The value ranges from 1 to 1440 minutes.

Default value: 2

Maximum value: 1440

Example

```
add lb wlm ibm_wlm 10.102.1.10 3060
```

rm lb wlm

Removes a Work Load Manager. NOTE: This command is deprecated. WLM feature has been deprecated from Kos onwards as classic build is not supported.

Synopsys

Arguments

wlmName

The name of the Work Load Manager to be removed.

Example

```
rm lb wlm ibm_wlm
```

set lb wlm

set Work Load Manager attributes NOTE: This command is deprecated.

Synopsys

Arguments

wlmName

The name of the work load manager.

KATimeout

The idle time period after which NS would probe the WLM. The value ranges from 1 to 1440 minutes.

Default value: 2

Maximum value: 1440

Example

```
set lb wlm ibm_wlm -ka_timeout 6
```

unset lb wlm

Use this command to remove lb wlm settings.Refer to the set lb wlm command for meanings of the arguments.NOTE: This command is deprecated.

Synopsys

show lb wlm

show Work Load Manager details NOTE: This command is deprecated.WLM feature has been deprecated from Kos onwards as classic build is not supported.

Synopsys

Arguments

wlmName

The name of the work load manager.

Outputs

IPAddress

The IP address of the WLM.

port

A port number for the virtual server.

stateflag

secure

Use this parameter to enable secure mode of communication with WLM.

KATimeout

The idle time period after which NS would probe the WLM. The value ranges from 1 to 1440 minutes.

LBUID

The LBUID for the Load Balancer to communicate to the Work Load Manager.

state

State of the WLM.

vServerName

Name of the virtual server which is to be bound to the WLM.

devno

count

Example

```
show lb wlm ibm_wlm
```

bind lb wlm

Bind a vserver to Work Load Manager. NOTE: This command is deprecated.WLM feature has been deprecated from Kos onwards as classic build is not supported.

Synopsys

Arguments

wlmName

The name of the Work Load Manager.

vServerName

Name of the virtual server which is to be bound to the WLM.

Example

```
bind lb wlm ibm_wlm http_vip To bind multiple vservers to workload manager use the follow
```

unbind lb wlm

Unbind a vserver from Work Load Manager. NOTE: This command is deprecated.WLM feature has been deprecated from Kos onwards as classic build is not supported.

Synopsys

Arguments

wlmName

The name of the Work Load Manager.

vServerName

Name of the virtual server which is to be unbound from the WLM.

Example

```
unbind lb wlm ibm_wlm http_vip To unbind multiple vservers from Work Load Manager use the
```

LLDP Commands

The entities on which you can perform NetScaler CLI operations:

- o lldp
- o lldp neighbors
- o lldp param
- o lldp stats

Ildp

The following operations can be performed on "Ildp":

stat Ildp

Display Ildp statistics.

Synopsys

stat Ildp [<ifnum>@] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

ifnum

LLDP Statistics per interfaces

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

LLDP received Packets (RxPkts)

Total LLDP Packets received.

LLDP bytes received (RxBytes)

Total LLDP bytes received

LLDP packets transmitted (TxPkts)

Total LLDP Packets transmitted

LLDP bytes transmitted (TxBytes)

Total LLDP bytes transmitted.

Errors in LLDP Packets (RxErrPkts)

Total errors in LLDP packets.

Discarded LLDP Packets (DiscardPkts)

Total discarded LLDP packets.

TLVs not Recognised (TlvNotRecognised)

Total TLVs not Recognised.

Ildp neighbors

The following operations can be performed on "Ildp neighbors":

[show](#) | [clear](#)

show Ildp neighbors

Display Neighbor information per interface

Synopsys

show Ildp neighbors [<ifnum>]

Arguments

ifnum

Interface Name

Outputs

chassisIdSubtype

Chassis id sub type

chassisId

Chassis Id

portIdSubtype

Port id subtype

portId

Port Id

TTL

Time to Live

portDescription

Port Description

sys

System Name

sysDesc

System Description

mgmtAddressSubtype

Management Address Type

mgmtAddress

Management Address

iftype

Interface subtype

ifNumber

Interface Number

vlan

vlan name

vlanId

Vlan Id

portProtoSupported

Flag to show Port Protocol Support

portProtoEnabled

Flag to show Port Protocol support enabled

portProtold

Port Protocol ID

portVlanId

Port Vlan Id

protocold

port vlan id name

linkAggrCapable

Is neighbor Link Aggregation Capable

LinkAggrEnabled

Is link aggregation Enabled

linkAggrId

Link Aggregation Id

flag

sysCapabilities

Acronyms for remote system capabilities:

OT : Other.

RE : Repeater(IETF RFC 2108).

BR : MAC Bridge(IEEE Std 802.1D).

WL : WLAN Access Point(IEEE Std 802.11 MIB).

RO : ROuter(IETF RFC 1812).

TE : Telephone(IETF RFC 4293).

DO : DOCSIS cable device(IETF RFC 4639 and IETF RFC 4546).

ST : STation only(IETF RFC 4293).

CV : C-VLAN component of a VLAN Bridge(IEEE Std 802.1Q).

SV : S-VLAN component of a VLAN Bridge(IEEE Std 802.1Q).

MR : Two port MAC Relay(IEEE Std 802.1Q).

sysCapEnabled

Acronyms for remote system enabled capabilities:

OT : Other.

RE : Repeater(IETF RFC 2108).

BR : MAC Bridge(IEEE Std 802.1D).

WL : WLAN Access Point(IEEE Std 802.11 MIB).

RO : Router(IETF RFC 1812).

TE : Telephone(IETF RFC 4293).

DO : DOCSIS cable device(IETF RFC 4639 and IETF RFC 4546).

ST : STation only(IETF RFC 4293).

CV : C-VLAN component of a VLAN Bridge(IEEE Std 802.1Q).

SV : S-VLAN component of a VLAN Bridge(IEEE Std 802.1Q).

MR : Two port MAC Relay(IEEE Std 802.1Q).

autonegSupport

MAC/PHY autonegotiation support

autonegEnabled

MAC/PHY autonegotiation enabled

autonegAdvertised

MAC/PHY autonegotiation advertised

autonegMAUType

MAC/PHY Medium Attachment Unit (MAU) type of the port. Description listed below:

AUI - no internal MAU, view from AUI

10Base5 - thick coax MAU

Foirl - FOIRL MAU

10Base2 - thin coax MAU

10BaseT - UTP MAU

10BaseFP - passive fiber MAU

10BaseFB - sync fiber MAU

10BaseFL - async fiber MAU

10Broad36 - broadband DTE MAU

10BaseTHD - UTP MAU, half duplex mode

10BaseTFD - UTP MAU, full duplex mode

10BaseFLHD - async fiber MAU, half duplex mode

10BaseFLDF - async fiber MAU, full duplex mode

10BaseT4 - 4 pair category 3 UTP

100BaseTXHD - 2 pair category 5 UTP, half duplex mode

100BaseTXFD - 2 pair category 5 UTP, full duplex mode

100BaseFXHD - X fiber over PMT, half duplex mode

100BaseFXFD - X fiber over PMT, full duplex mode

100BaseT2HD - 2 pair category 3 UTP, half duplex mode

100BaseT2DF - 2 pair category 3 UTP, full duplex mode

1000BaseXHD - PCS/PMA, unknown PMD, half duplex mode

1000BaseXFD - PCS/PMA, unknown PMD, full duplex mode

1000BaseLXHD - Fiber over long-wavelength laser, half duplex mode

1000BaseLXFD - Fiber over long-wavelength laser, full duplex mode

1000BaseSXHD - Fiber over short-wavelength laser, half duplex mode

1000BaseSXFD - Fiber over short-wavelength laser, full duplex mode

1000BaseCXHD - Copper over 150-Ohm balanced cable, half duplex mode

1000BaseCXFD - Copper over 150-Ohm balanced cable, full duplex mode

1000BaseTHD - Four-pair Category 5 UTP, half duplex mode

1000BaseTFD - Four-pair Category 5 UTP, full duplex mode

10GigBaseX - X PCS/PMA, unknown PMD

10GigBaseLX4 - X fiber over WWDM optics

10GigBaseR - R PCS/PMA, unknown PMD

10GigBaseER - R fiber over 1550 nm optics

10GigBaseLR - R fiber over 1310 nm optics

10GigBaseSR - R fiber over 850 nm optics

10GigBaseW - W PCS/PMA, unknown PMD

10GigBaseEW - W fiber over 1550 nm optics

10GigBaseLW - W fiber over 1310 nm optics

10GigBaseSW - W fiber over 850 nm optics

mtu

MTU of Remote Device

devno

count

stateflag

clear lldp neighbors

Removes LLDP neighbor info of interfaces

Synopsys

clear lldp neighbors

Ildp param

The following operations can be performed on "Ildp param":

[set](#) | [unset](#) | [show](#)

set Ildp param

Sets the global Link Layer Discovery Protocol (LLDP) parameters such as LLDP Timer, Hold Time Multiplier, and LLDP mode.

Synopsys

```
set Ildp param [-holdtimeTxMult <positive_integer>] [-timer <positive_integer>] [-Mode <Mode>]
```

Arguments

holdtimeTxMult

A multiplier for calculating the duration for which the receiving device stores the LLDP information in its database before discarding or removing it. The duration is calculated as the holdtimeTxMult (Holdtime Multiplier) parameter value multiplied by the timer (Timer) parameter value.

Default value: 4

Minimum value: 1

Maximum value: 20

timer

Interval, in seconds, between LLDP packet data units (LLDPDUs). that the NetScaler ADC sends to a directly connected device.

Default value: 30

Minimum value: 1

Maximum value: 3000

Mode

Global mode of Link Layer Discovery Protocol (LLDP) on the NetScaler ADC. The resultant LLDP mode of an interface depends on the LLDP mode configured at the global and the interface levels.

Possible values: NONE, TRANSMITTER, RECEIVER, TRANSCEIVER

Example

```
set lldpparam -mode RECEIVER
```

unset Ildp param

Use this command to remove Ildp param settings. Refer to the set Ildp param command for meanings of the arguments.

Synopsys

```
unset Ildp param [-holdtimeTxMult] [-timer] [-Mode]
```

show Ildp param

Display the global LLDP params

Synopsys

show lldp param

Outputs

holdtimeTxMult

A multiplier for calculating the duration for which the receiving device stores the LLDP information in its database before discarding or removing it. The duration is calculated as the holdtimeTxMult (Holdtime Multiplier) parameter value multiplied by the timer (Timer) parameter value.

timer

Interval, in seconds, between LLDP packet data units (LLDPDUs). that the NetScaler ADC sends to a directly connected device.

Mode

Global mode of Link Layer Discovery Protocol (LLDP) on the NetScaler ADC. The resultant LLDP mode of an interface depends on the LLDP mode configured at the global and the interface levels.

Example

show lldpparam

Ildp stats

The following operations can be performed on "Ildp stats":

show Ildp stats

show Ildp stats is an alias for stat Ildp Display LLDP stats

Synopsys

show Ildp stats - alias for 'stat Ildp'

Networking Commands

The entities on which you can perform NetScaler CLI operations:

- o L2Param
- o L3Param
- o L4Param
- o arp
- o arpparam
- o bridge
- o bridgegroup
- o bridgetable
- o channel
- o ci
- o fis
- o forwardingSession
- o inat
- o inatparam
- o inatsession
- o interface
- o interfacePair
- o ip6Tunnel
- o ip6TunnelParam
- o ipTunnel
- o ipTunnelParam
- o ipset
- o ipv6
- o lacp
- o linkset
- o nat64
- o nd6
- o nd6RAvariables
- o netProfile
- o netbridge
- o onLinkIPv6Prefix
- o ptp
- o rnat
- o rnat6
- o rnatglobal
- o rnatip
- o rnatparam
- o route
- o route6
- o rsskeytype
- o tunnelip
- o tunnelip6
- o vPathParam
- o vlan
- o vpath
- o vrID
- o vrID6
- o vrIDParam
- o vxlan

L2Param

The following operations can be performed on "L2Param":

[set](#) | [unset](#) | [show](#)

set L2Param

Set Layer 2 related global settings on the NetScaler

Synopsys

```
set L2Param [-mbfPeermacUpdate <positive_integer>] [-maxBridgeCollision <positive_integer>] [-bdggrpProxyArp (
ENABLED | DISABLED )] [-bdgSetting ( ENABLED | DISABLED )] [-garpOnVridIntf ( ENABLED | DISABLED )] [-
macModeFwdMyPkt ( ENABLED | DISABLED )] [-useMyMAC ( ENABLED | DISABLED )] [-proxyArp ( ENABLED |
DISABLED )] [-garpReply ( ENABLED | DISABLED )] [-mbfInstLearning ( ENABLED | DISABLED )] [-rstIntfOnHaFo
( ENABLED | DISABLED )] [-skipProxyingBsdTraffic ( ENABLED | DISABLED )] [-returnToEthernetSender (
ENABLED | DISABLED )] [-stopMacMoveUpdate ( ENABLED | DISABLED )]
```

Arguments

mbfPeermacUpdate

When mbf_instant_learning is enabled, learn any changes in peer's MAC after this time interval, which is in 10ms ticks.

Default value: 10

Minimum value: 0

maxBridgeCollision

Maximum bridge collision for loop detection

Default value: 20

Minimum value: 0

bdggrpProxyArp

Set/reset proxy ARP in bridge group deployment

Possible values: ENABLED, DISABLED

Default value: ENABLED

bdgSetting

Bridging settings for C2C behavior

Possible values: ENABLED, DISABLED

Default value: DISABLED

garpOnVridIntf

Send GARP messages on VRID-configured interfaces upon failover

Possible values: ENABLED, DISABLED

Default value: ENABLED

macModeFwdMyPkt

MAC mode vserver forward packets destined to VIPs.

Possible values: ENABLED, DISABLED

Default value: DISABLED

useMyMAC

Set/reset `cfg_use_my_mac`

Possible values: ENABLED, DISABLED

Default value: DISABLED

proxyArp

Set/reset `cfg_proxy_arp_dr`

Possible values: ENABLED, DISABLED

Default value: ENABLED

garpReply

Set/reset REPLY form of GARP

Possible values: ENABLED, DISABLED

Default value: DISABLED

mbfInstLearning

Enable instant learning of MAC changes in MBF mode.

Possible values: ENABLED, DISABLED

Default value: DISABLED

rstIntfOnHaFo

Enable the reset interface upon HA failover.

Possible values: ENABLED, DISABLED

Default value: DISABLED

skipProxyingBsdTraffic

Enable the proxying of FreeBSD traffic.

Possible values: ENABLED, DISABLED

Default value: DISABLED

returnToEthernetSender

Return to ethernet sender.

Possible values: ENABLED, DISABLED

Default value: DISABLED

stopMacMoveUpdate

Stop propagation of server mac change to natpcbs

Possible values: ENABLED, DISABLED

Default value: DISABLED

unset L2Param

Use this command to remove L2Param settings. Refer to the set L2Param command for meanings of the arguments.

Synopsys

unset L2Param [-mbfPeermacUpdate] [-maxBridgeCollision] [-bdggrpProxyArp] [-bdgSetting] [-garpOnVridIntf] [-macModeFwdMyPkt] [-useMyMAC] [-proxyArp] [-garpReply] [-mbfInstLearning] [-rstIntfOnHaFo] [-skipProxyingBsdTraffic] [-returnToEthernetSender] [-stopMacMoveUpdate]

show L2Param

Displays the settings of global Layer 2 parameters on the NetScaler appliance.

Synopsys

show L2Param

Outputs

maxBridgeCollision

Maximum bridge collision for loop detection

linkMTU

this MTU is used for Ready logo purpose, changing the Interface MTU at soft layer level.

mbfPeermacUpdate

When mbf_instant_learning is enabled, learn any changes in peer's MAC after this time interval, which is in 10ms ticks.

bdggrpProxyArp

Set/reset proxy ARP in bridge group deployment

bdgSetting

Bridging settings for C2C behavior

garpOnVridIntf

Send GARP messagess on VRID-configured interfaces upon failover

macModeFwdMyPkt

MAC mode vsrver forward packets destined to VIPs.

useMyMAC

Set/reset cfg_use_my_mac

proxyArp

Set/reset cfg_proxy_arp_dr

garpReply

Set/reset REPLY form of GARP

mbfInstLearning

Enable instant learning of MAC changes in MBF mode.

rstIntfOnHaFo

Enable the reset interface upon HA failover.

skipProxyingBsdTraffic

Enable the proxying of FreeBSD traffic.

returnToEthernetSender

Return to ethernet sender.

stopMacMoveUpdate

Stop propagation of server mac change to natpcbs

L3Param

The following operations can be performed on "L3Param":

[set](#) | [unset](#) | [show](#)

set L3Param

Set Layer 3 related global settings on the NetScaler

Synopsys

```
set L3Param [-srcnat ( ENABLED | DISABLED )] [-icmpGenRateThreshold <positive_integer>] [-overrideRnat (
ENABLED | DISABLED )] [-dropDFFlag ( ENABLED | DISABLED )] [-mipRoundRobin ( ENABLED | DISABLED )] [-
externalLoopBack ( ENABLED | DISABLED )] [-tnlPmtuWoConn ( ENABLED | DISABLED )] [-usipServerStrayPkt (
ENABLED | DISABLED )] [-forwardICMPFragments ( ENABLED | DISABLED )] [-dropIPFragments ( ENABLED |
DISABLED )] [-AclLogTime <positive_integer>] [-icmpErrGenerate ( ENABLED | DISABLED )] [-implicitACLAllow (
ENABLED | DISABLED )]
```

Arguments

srcnat

Perform NAT if only the source is in the private network

Possible values: ENABLED, DISABLED

Default value: ENABLED

icmpGenRateThreshold

NS generated ICMP pkts per 10ms rate threshold

Default value: 100

Minimum value: 0

overrideRnat

USNIP/USIP settings override RNAT settings for configured
service/virtual server traffic..

Possible values: ENABLED, DISABLED

Default value: DISABLED

dropDFFlag

Enable dropping the IP DF flag.

Possible values: ENABLED, DISABLED

Default value: DISABLED

mipRoundRobin

Enable round robin usage of mapped IPs.

Possible values: ENABLED, DISABLED

Default value: ENABLED

externalLoopBack

Enable external loopback.

Possible values: ENABLED, DISABLED

Default value: DISABLED

tnlPmtuWoConn

Enable external loopback.

Possible values: ENABLED, DISABLED

Default value: ENABLED

usipServerStrayPkt

Enable detection of stray server side pkts in USIP mode.

Possible values: ENABLED, DISABLED

Default value: DISABLED

forwardICMPFragments

Enable forwarding of ICMP fragments.

Possible values: ENABLED, DISABLED

Default value: DISABLED

dropIPFragments

Enable dropping of IP fragments.

Possible values: ENABLED, DISABLED

Default value: DISABLED

AcLogTime

Parameter to tune acl logging time

Default value: 5000

Minimum value: 0

icmpErrGenerate

Enable/Disable fragmentation required icmp error generation, before encapsulating a packet with vPath header. This knob is only functional for vPath Environment

Possible values: ENABLED, DISABLED

Default value: ENABLED

implicitACLAllow

Do not apply ACLs for internal ports

Possible values: ENABLED, DISABLED

Default value: ENABLED

unset L3Param

Use this command to remove L3Param settings. Refer to the set L3Param command for meanings of the arguments.

Synopsys

```
unset L3Param [-srcnat] [-icmpGenRateThreshold] [-overrideRnat] [-dropDFFlag] [-mipRoundRobin] [-externalLoopBack] [-tnlPmtuWoConn] [-usipServerStrayPkt] [-forwardICMPFragments] [-dropIPFragments] [-AcLogTime] [-icmpErrGenerate] [-implicitACLAllow]
```

show L3Param

Displays the settings of global Layer 3 parameters.

Synopsys

show L3Param

Outputs

srcnat

Perform NAT if only the source is in the private network

icmpGenRateThreshold

NS generated ICMP pkts per 10ms rate threshold

overrideRnat

USNIP/USIP settings override RNAT settings for configured service/virtual server traffic..

dropDFFlag

Enable dropping the IP DF flag.

mipRoundRobin

Enable round robin usage of mapped IPs.

externalLoopBack

Enable external loopback.

tnIPmtuWoConn

Enable external loopback.

usipServerStrayPkt

Enable detection of stray server side pkts in USIP mode.

forwardICMPFragments

Enable forwarding of ICMP fragments.

dropIPFragments

Enable dropping of IP fragments.

AcLogTime

Parameter to tune acl logging time

icmpErrGenerate

Enable/Disable fragmentation required icmp error generation, before encapsulating a packet with vPath header. This knob is only functional for vPath Environment

implicitACLAllow

Do not apply ACLs for internal ports

L4Param

The following operations can be performed on "L4Param":

[set](#) | [unset](#) | [show](#)

set L4Param

Set Layer 4 related global settings on the NetScaler

Synopsys

```
set L4Param [-I2ConnMethod <I2ConnMethod>] [-I4switch ( ENABLED | DISABLED )]
```

Arguments

I2ConnMethod

Layer 2 connection method based on the combination of channel number, MAC address and VLAN. It is tuned with I2conn param of lb vserver. If I2conn of lb vserver is ON then method specified here will be used to identify a connection in addition to the 4-tuple (<source IP>:<source port>:<destination IP>:<destination port>).

Possible values: Channel, Vlan, VlanChannel, Mac, MacChannel, MacVlan, MacVlanChannel

Default value: MacVlanChannel

I4switch

In L4 switch topology, always clients and servers are on the same side. Enable I4switch to allow such connections.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
set l4param
```

unset L4Param

Use this command to remove L4Param settings. Refer to the set L4Param command for meanings of the arguments.

Synopsys

```
unset L4Param [-I2ConnMethod] [-I4switch]
```

show L4Param

Displays the settings of global Layer 4 parameters.

Synopsys

```
show L4Param
```

Outputs

I2ConnMethod

Layer 2 connection method based on the combination of channel number, MAC address and VLAN. It is tuned with l2conn param of lb vserver. If l2conn of lb vserver is ON then method specified here will be used to identify a connection in addition to the 4-tuple (<source IP>:<source port>::<destination IP>:<destination port>).

l4switch

In L4 switch topology, always clients and servers are on the same side. Enable l4switch to allow such connections.

arp

The following operations can be performed on "arp":

add | **rm** | **send** | **show**

add arp

Adds a static ARP entry to the ARP table of the NetScaler appliance.

Synopsys

```
add arp -IPAddress <ip_addr> [-td <positive_integer>] -mac <mac_addr> (-ifnum <interface_name> | (-vxlان <positive_integer> -vtep <ip_addr>)) [-ownerNode <positive_integer>]
```

Arguments

IPAddress

IP address of the network device that you want to add to the ARP table.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

mac

MAC address of the network device.

ifnum

Interface through which the network device is accessible. Specify the interface in (slot/port) notation. For example, 1/3.

vxlان

ID of the VXLAN on which the IP address of this ARP entry is reachable.

Minimum value: 1

Maximum value: 16777215

vtep

IP address of the VXLAN tunnel endpoint (VTEP) through which the IP address of this ARP entry is reachable.

ownerNode

The owner node for the Arp entry.

Default value: -1

Minimum value: 0

Maximum value: 31

Example

```
add arp -ip 10.100.0.48 -mac 00:a0:cc:5f:76:3a -ifnum 1/1
```

rm arp

Removes a specified static ARP entry or all static ARP entries from the NetScaler appliance's ARP table.

Synopsys

```
rm arp (<IPAddress> | -all) [-td <positive_integer>] [-ownerNode <positive_integer>]
```

Arguments

IPAddress

IP address of the network device in the ARP entry that you want to remove from the ARP table.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

all

Remove all ARP entries from the ARP table of the NetScaler appliance.

ownerNode

The owner node for the Arp entry.

Default value: -1

Minimum value: 0

Maximum value: 31

send arp

Sends Gratuitous Address Resolution Protocol (GARP) messages for the specified NetScaler owned IP addresses.

Synopsys

```
send arp ((-IPAddress <ip_addr> [-td <positive_integer>]) | -all)
```

Arguments

IPAddress

NetScaler owned IP address for which the NetScaler appliance sends Gratuitous Address Resolution Protocol (GARP) messages.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

all

Send GARP messages for all NetScaler owned IP addresses on which the ARP option is enabled. In a secondary node of an high availability configuration, this option sends GARP messages for the node's NSIP address only.

Example

```
send arp 10.10.10.10
```

show arp

Display all the entries in the system's ARP table.

Synopsys

```
show arp [<IPAddress> [-td <positive_integer>] [-ownerNode <positive_integer>]]
```

Arguments

IPAddress

The IP address corresponding to an ARP entry.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

ownerNode

The cluster node which owns the ARP entry.

Default value: -1

Minimum value: 0

Maximum value: 31

Outputs

mac

The MAC address corresponding to an ARP entry.

ifnum

The interface on which this MAC address resides.

timeout

The time, in seconds, after which the entry times out.

state

The state of the ARP entry.

flags

The flags for the entry.

type

Indicates whether this ARP entry was added manually or dynamically. When you manually add an ARP entry, the value for this parameter is STATIC. Otherwise, it is DYNAMIC. For the NSIP and loopback IP addresses, the value is PERMANENT.

vlan

The VLAN ID through which packets are to be sent after matching the ARP entry. This is a numeric value.

vxlan

ID of the VXLAN on which the IP address of this ARP entry is reachable.

vtep

IP address of the VXLAN tunnel endpoint (VTEP) through which the IP address of this ARP entry is reachable.

channel

The tunnel, channel, or physical interface through which the ARP entry is identified.

flag

Flags for the entry.

devno**count****stateflag**

Example

The output of the `sh arp` command is as follows: 5 configured arps: IP

MAC

arpparam

The following operations can be performed on "arpparam":

[set](#) | [unset](#) | [show](#)

set arpparam

Sets a global time-out value for dynamic ARP entries.

Synopsis

```
set arpparam [-timeout <positive_integer>] [-spoofValidation ( ENABLED | DISABLED )]
```

Arguments

timeout

Time-out value (aging time) for the dynamically learned ARP entries, in seconds. The new value applies only to ARP entries that are dynamically learned after the new value is set. Previously existing ARP entries expire after the previously configured aging time.

Default value: 1200

Minimum value: 5

Maximum value: 1200

spoofValidation

enable/disable arp spoofing validation

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
set arpparam -timeout 200 -spoofvalidate ENABLE
```

unset arpparam

Use this command to remove arpparam settings. Refer to the set arpparam command for meanings of the arguments.

Synopsis

```
unset arpparam [-timeout] [-spoofValidation]
```

show arpparam

Display the global setting of dynamically learned ARP entries.

Synopsis

```
show arpparam
```

Outputs

timeout

The ARP table entry aging time, in seconds.

spoofValidation

enable/disable arp spoofing validation

Example

```
show arpparam
```

bridge

The following operations can be performed on "bridge":

stat bridge

Display bridging statistics.

Synopsys

```
stat bridge [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

Loops

The number of times bridging registered MAC moved

Collisions (Collisns)

The number of bridging table collisions

Interface muted (Mutes)

The number of bridging related interface mutes

Total bridged packets (Tot_pkts)

The total number of bridged packets

Total bridged Mbits (Tot_Mbits)

The total number of bridged Mbits

bridgegroup

The following operations can be performed on "bridgegroup":

add | **rm** | **set** | **unset** | **bind** | **unbind** | **show**

add bridgegroup

Create a Bridge group.

Synopsys

```
add bridgegroup <id> [-ipv6DynamicRouting ( ENABLED | DISABLED )]
```

Arguments

id

An integer that uniquely identifies the bridge group.

Minimum value: 1

Maximum value: 1000

ipv6DynamicRouting

Enable all IPv6 dynamic routing protocols on all VLANs bound to this bridgegroup. Note: For the ENABLED setting to work, you must configure IPv6 dynamic routing protocols from the VTYSH command line.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
add bridgegroup 1
```

rm bridgegroup

Remove the bridge group created by the add bridge group command.

Synopsys

```
rm bridgegroup <id>
```

Arguments

id

An integer that uniquely identifies the bridge group that you want to remove from the NetScaler appliance.

Minimum value: 1

Maximum value: 1000

set bridgegroup

Set Bridge group parameters.

Synopsys

```
set bridgegroup <id> -ipv6DynamicRouting ( ENABLED | DISABLED )
```

Arguments

id

An integer value that uniquely identifies the bridge group. Minimum value: 1. Maximum value: 1000.

Minimum value: 1

Maximum value: 1000

ipv6DynamicRouting

Enable all IPv6 dynamic routing protocols on this bridge group. For this setting to work, you must configure IPv6 dynamic routing protocols from the VTYSH command line. For more information about configuring IPv6 dynamic routing protocols on the NetScaler appliance, see the Dynamic Routing chapter of the Citrix NetScaler Networking Guide.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
set bridgegroup 1 -dynamicRouting ENABLED
```

unset bridgegroup

Use this command to remove bridgegroup settings. Refer to the set bridgegroup command for meanings of the arguments.

Synopsis

```
unset bridgegroup <id> -ipv6DynamicRouting
```

bind bridgegroup

Bind a vlan or an ip address to a bridgegroup.

Synopsis

```
bind bridgegroup <id> [-vlan <positive_integer>] [-IPAddress <ip_addr|ipv6_addr|*> [<netmask>] [-td <positive_integer>]]
```

Arguments

id

The integer that uniquely identifies the bridge group.

Minimum value: 1

Maximum value: 1000

vlan

An integer that uniquely identifies the VLAN that you want to bind to this bridge group.

Minimum value: 2

Maximum value: 4094

IPAddress

A network address or addresses to be associated with the bridge group. You must add entries for these network addresses in the routing table before running this command.

netmask

A subnet mask associated with the network address.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

Example

```
bind bridgegroup 1 -vlan 2
```

unbind bridgegroup

Unbinds the specified VLANs or IP addresses from a bridge group.

Synopsys

```
unbind bridgegroup <id> [-vlan <positive_integer>] [-IPAddress <ip_addr|ipv6_addr*> [<netmask>] [-td  
<positive_integer>]]
```

Arguments

id

Integer that uniquely identifies the bridge group.

Minimum value: 1

Maximum value: 1000

vlan

ID of the VLAN to unbind from this bridge group.

Minimum value: 2

Maximum value: 4094

IPAddress

Network address associated with the bridge group.

netmask

The network mask for the subnet defined for the bridge group.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

show bridgegroup

Display the configured bridge group. If a name is specified, only that particular bridge group information is displayed. Otherwise, all configured bridge groups are displayed.

Synopsys

show bridgegroup [<id>]

Arguments

id

The name of the bridge group.

Minimum value: 1

Maximum value: 1000

Outputs

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

IPAddress

The IP address assigned to the bridge group.

netmask

The network mask for the subnet defined for the bridge group.

flags

Temporary flag used for internal purpose.

portbitmap

Member interfaces of this bridge group.

tagbitmap

Tagged members of this bridge group.

ifaces

Names of all member interfaces of this bridge group.

taglfaces

Names of all tagged member interfaces of this bridge group.

vlan

Names of all member VLANs.

ipv6DynamicRouting

Whether dynamic routing is enabled or disabled.

rnat

Temporary flag used for internal purpose.

flag

devno

count

stateflag

Example

An example of the output of the show bridge group command is as follows: 2 configured Bri

bridgetable

The following operations can be performed on "bridgetable":

`set` | `unset` | `show` | `clear`

set bridgetable

Sets global parameters of bridge table entries.

Synopsys

```
set bridgetable -bridgeAge <positive_integer>
```

Arguments

bridgeAge

Time-out value for the bridge table entries, in seconds. The new value applies only to the entries that are dynamically learned after the new value is set. Previously existing bridge table entries expire after the previously configured time-out value.

Default value: 300

Minimum value: 60

Maximum value: 300

Example

```
set bridgetable -bridgeAge 200
```

unset bridgetable

Use this command to remove bridgetable settings. Refer to the set bridgetable command for meanings of the arguments.

Synopsys

```
unset bridgetable -bridgeAge
```

show bridgetable

Displays the bridge table entries and the configured time-out values for these entries.

Synopsys

```
show bridgetable
```

Outputs

bridgeAge

Time-out value for the bridge table entries, in seconds. The new value applies only to the entries that are dynamically learned after the new value is set. Previously existing bridge table entries expire after the previously configured time-out value.

mac

The MAC address of the target.

ifnum

The interface on which the address was learned.

vlan

The VLAN in which this MAC address resides.

vxlan

The VXLAN in which this MAC address resides.

vtep

The IP address of the VTEP

flags

Display flags,

channel

The Tunnel through which bridge entry is learned.

devno**count****stateflag**

Example

```
show bridgetable
```

clear bridgetable

Remove entries from bridge table

Synopsys

```
clear bridgetable [-vlan <positive_integer> | -vxlan <positive_integer>] [-ifnum <interface_name>]
```

Arguments

vlan

VLAN whose entries are to be removed.

Minimum value: 1

Maximum value: 4094

ifnum

INTERFACE whose entries are to be removed.

vxlan

VXLAN whose entries are to be removed.

Minimum value: 1

Maximum value: 16777215

channel

The following operations can be performed on "channel":

add | **rm** | **set** | **unset** | **bind** | **unbind** | **show**

add channel

Creates a link aggregate channel on the NetScaler appliance or on a cluster configuration. Link aggregation combines data coming from multiple ports into a single high-speed link. Configuring link aggregation increases the capacity and availability of the communication channel between the NetScaler appliance and other connected devices. When a network interface is bound to a channel, the channel parameters have precedence over the network interface parameters. That is, the network interface parameters are ignored. A network interface can be bound only to one channel.

Synopsys

```
add channel <id> [-ifnum <interface_name> ...] [-state ( ENABLED | DISABLED )] [-lmac <mac_addr>] [-speed <speed>] [-flowControl <flowControl>] [-haMonitor ( ON | OFF )] [-tagall ( ON | OFF )] [-ifAlias <string>] [-throughput <positive_integer>] [-bandwidthHigh <positive_integer>] [-bandwidthNormal <positive_integer>]]
```

Arguments

id

ID for the LA channel or cluster LA channel to be created. Specify an LA channel in LA/x notation, where x can range from 1 to 8 or cluster LA channel in CLA/x notation, where x can range from 1 to 4. Cannot be changed after the LA channel is created.

ifnum

Interfaces to be bound to the LA channel of a NetScaler appliance or to the LA channel of a cluster configuration.

For an LA channel of a NetScaler appliance, specify an interface in C/U notation (for example, 1/3).

For an LA channel of a cluster configuration, specify an interface in N/C/U notation (for example, 2/1/3).

where C can take one of the following values:

* 0 - Indicates a management interface.

* 1 - Indicates a 1 Gbps port.

* 10 - Indicates a 10 Gbps port.

U is a unique integer for representing an interface in a particular port group.

N is the ID of the node to which an interface belongs in a cluster configuration.

Use spaces to separate multiple entries.

state

Enable or disable the LA channel.

Possible values: ENABLED, DISABLED

Default value: ENABLED

lmac

Specifies a MAC address for the LA channels configured in NetScaler virtual appliances (VPX). This MAC address is persistent after each reboot. If you don't specify this parameter, a MAC address is generated randomly for each LA channel. These MAC addresses changes after each reboot.

speed

Ethernet speed of the channel, in Mbps. If the speed of any bound interface is greater than or equal to the value set for this parameter, the state of the interface is UP. Otherwise, the state is INACTIVE. Bound interfaces whose state is INACTIVE do not process any traffic.

Possible values: AUTO, 10, 100, 1000, 10000, 40000

Default value: AUTO

flowControl

Specifies the flow control type for this LA channel to manage the flow of frames. Flow control is a function as mentioned in clause 31 of the IEEE 802.3 standard. Flow control allows congested ports to pause traffic from the peer device. Flow control is achieved by sending PAUSE frames.

Possible values: OFF, RX, TX, RXTX

Default value: OFF

haMonitor

In a High Availability (HA) configuration, monitor the LA channel for failure events. Failure of any LA channel that has HA MON enabled triggers HA failover.

Possible values: ON, OFF

Default value: ON

tagall

Adds a four-byte 802.1q tag to every packet sent on this channel. The ON setting applies tags for all VLANs that are bound to this channel. OFF applies the tag for all VLANs other than the native VLAN.

Possible values: ON, OFF

Default value: OFF

ifAlias

Alias name for the LA channel. Used only to enhance readability. To perform any operations, you have to specify the LA channel ID.

Default value: " "

throughput

Low threshold value for the throughput of the LA channel, in Mbps. In an high availability (HA) configuration, failover is triggered when the LA channel has HA MON enabled and the throughput is below the specified threshold.

Minimum value: 0

Maximum value: 160000

bandwidthHigh

High threshold value for the bandwidth usage of the LA channel, in Mbps. The NetScaler appliance generates an SNMP trap message when the bandwidth usage of the LA channel is greater than or equal to the specified high threshold value.

Minimum value: 0

Maximum value: 160000

bandwidthNormal

Normal threshold value for the bandwidth usage of the LA channel, in Mbps. When the bandwidth usage of the LA channel returns to less than or equal to the specified normal threshold after exceeding the high threshold, the NetScaler appliance generates an SNMP trap message to indicate that the bandwidth usage has returned to normal.

Minimum value: 0

Maximum value: 160000

rm channel

Removes an LA channel from the NetScaler appliance or a cluster LA channel from a cluster configuration. Important: When a LA channel is removed, the network interfaces bound to it induce network loops that decrease network performance. You must disable the network interfaces before you remove the channel.

Synopsys

```
rm channel <id>
```

Arguments

id

ID of the LA channel or cluster LA channel that you want to remove. Specify an LA channel in LA/x notation, where x can range from 1 to 8 or a cluster LA channel in CLA/x notation, where x can range from 1 to 4.

set channel

Modifies the specified parameters of an LA channel.

Synopsys

```
set channel <id> [-state ( ENABLED | DISABLED )] [-lmac <mac_addr>] [-speed <speed>] [-mtu  
<positive_integer>] [-flowControl <flowControl>] [-haMonitor ( ON | OFF )] [-tagall ( ON | OFF )] [-ifAlias <string>] [-  
throughput <positive_integer>] [-lrMinThroughput <positive_integer>] [-linkRedundancy ( ON | OFF )] [-  
bandwidthHigh <positive_integer>] [-bandwidthNormal <positive_integer>]]
```

Arguments

id

ID of the LA channel or the cluster LA channel whose parameters you want to modify. Specify an LA channel in LA/x notation, where x can range from 1 to 8 or a cluster LA channel in CLA/x notation, where x can range from 1 to 4. Required for identifying the LA channel and cannot be modified.

state

Enable or disable the LA channel.

Possible values: ENABLED, DISABLED

Default value: ENABLED

lmac

Allows User to set MAC address for LA channels on Hypervised platforms.

speed

The speed for the LA channel.

Possible values: AUTO, 10, 100, 1000, 10000, 40000

Default value: AUTO

mtu

The maximum transmission unit (MTU) is the largest packet size, measured in bytes excluding 14 bytes ethernet header and 4 bytes crc, that can be transmitted and received by this interface. Default value of MTU is 1500 on all the interface of Netscaler appliance any value configured more than 1500 on the interface will make the interface as jumbo enabled. In case of cluster backplane interface MTU value will be changed to 1514 by default, user has to change the backplane interface value to maximum mtu configured on any of the interface in cluster system plus 14 bytes more for backplane interface if Jumbo is enabled on any of the interface in a cluster system. Changing the backplane will bring back the MTU of backplane interface to

default value of 1500. If a channel is configured as backplane then the same holds true for channel as well as member interfaces. In case of channel if member interfaces is configured as different mtu then the highest MTU configured MTU is treated as the LA MTU if MTU is not specified on LA explicitly. Low MTU interfaces in channel will be taken out of LA distribution list.

Default value: 1500

Minimum value: 1500

Maximum value: 9216

flowControl

Required flow control for the LA channel.

Possible values: OFF, RX, TX, RXTX

Default value: OFF

haMonitor

The state of HA monitoring for the LA channel.

Possible values: ON, OFF

Default value: ON

tagall

The appliance adds a four-byte 802.1q tag to every packet sent on this channel. ON applies tags for all the VLANs that are bound to this channel. OFF, applies the tag for all VLANs other than the native VLAN.

Possible values: ON, OFF

Default value: OFF

ifAlias

The alias name for the interface.

Default value: " "

throughput

Low threshold value for the throughput of the LA channel, in Mbps. In an high availability (HA) configuration, failover is triggered when the LA channel has HA MON enabled and the throughput is below the specified threshold.

Minimum value: 0

Maximum value: 160000

lrMinThroughput

Specifies the minimum throughput threshold (in Mbps) to be met by the active subchannel. Setting this parameter automatically divides an LACP channel into logical subchannels, with one subchannel active and the others in standby mode. When the maximum supported throughput of the active channel falls below the lrMinThroughput value, link failover occurs and a standby subchannel becomes active.

Minimum value: 0

Maximum value: 80000

linkRedundancy

Link Redundancy for Cluster LAG.

Possible values: ON, OFF

Default value: OFF

bandwidthHigh

High threshold value for the bandwidth usage of the LA channel, in Mbps. The NetScaler appliance generates an SNMP trap message when the bandwidth usage of the LA channel is greater than or equal to the specified high threshold value.

Minimum value: 0

Maximum value: 160000

bandwidthNormal

Normal threshold value for the bandwidth usage of the LA channel, in Mbps. When the bandwidth usage of the LA channel returns to less than or equal to the specified normal threshold after exceeding the high threshold, the NetScaler appliance generates an SNMP trap message to indicate that the bandwidth usage has returned to normal.

Minimum value: 0

Maximum value: 160000

unset channel

Use this command to remove channel settings. Refer to the set channel command for meanings of the arguments.

Synopsis

```
unset channel <id> [-state] [-speed] [-mtu] [-flowControl] [-haMonitor] [-tagall] [-ifAlias] [-throughput] [-lrMinThroughput] [-linkRedundancy] [-bandwidthHigh] [-bandwidthNormal]
```

bind channel

Binds the specified interfaces to a channel.

Synopsis

```
bind channel <id> <ifnum> ...
```

Arguments

id

ID of the LA channel or the cluster LA channel to which you want to bind interfaces. Specify an LA channel in LA/x notation, where x can range from 1 to 8 or a cluster LA channel in CLA/x notation, where x can range from 1 to 4.

ifnum

Interfaces to be bound to the LA channel of a NetScaler appliance or to the LA channel of a cluster configuration.

For an LA channel of a NetScaler appliance, specify an interface in C/U notation (for example, 1/3).

For an LA channel of a cluster configuration, specify an interface in N/C/U notation (for example, 2/1/3).

where C can take one of the following values:

* 0 - Indicates a management interface.

* 1 - Indicates a 1 Gbps port.

* 10 - Indicates a 10 Gbps port.

U is a unique integer for representing an interface in a particular port group.

N is the ID of the node to which an interface belongs in a cluster configuration.

Use spaces to separate multiple entries.

unbind channel

Unbinds the specified interfaces from an LA channel.

Synopsys

```
unbind channel <id> <ifnum> ...
```

Arguments

id

ID of the LA channel or cluster LA channel from which you want to unbind interfaces. Specify an LA channel in LA/x notation, where x can range from 1 to 8 or a cluster LA channel in CLA/x notation, where x can range from 1 to 4.

ifnum

Interfaces to be unbound from the LA channel of a NetScaler appliance or from the LA channel of a cluster configuration.

For an LA channel of a NetScaler appliance, specify an interface in C/U notation (for example, 1/3).

For an LA channel of a cluster configuration, specify an interface in N/C/U notation (for example, 2/1/3).

where C can take one of the following values:

* 0 - Indicates a management interface.

* 1 - Indicates a 1 Gbps port.

* 10 - Indicates a 10 Gbps port.

U is a unique integer for representing an interface in a particular port group.

N is the ID of the node to which an interface belongs in a cluster configuration.

Use spaces to separate multiple entries.

show channel

Displays the settings of all LA channels or of the specified channel. To display the settings of all channels, run the command without any parameters. To display the settings of a particular channel, specify the ID of the channel.

Synopsys

```
show channel [<id>]
```

Arguments

id

ID of an LA channel or LA channel in cluster configuration whose details you want the NetScaler appliance to display.

Specify an LA channel in LA/x notation, where x can range from 1 to 8 or a cluster LA channel in CLA/x notation, where x can range from 1 to 4.

Minimum value: 1

Outputs

stateflag

deviceName

LA channel name in form LA/x, where x is channel ID, which ranges from 1 to 8.

unit

Unit number of the channel. This is an internal reference number that the NetScaler uses to identify the channel.

description

The IEEE standard that the channel is based on.

flags

Flags of this channel.

mtu

The maximum transmission unit (MTU) is the largest packet size, measured in bytes excluding 14 bytes ethernet header and 4 bytes crc, that can be transmitted and received by this interface. Default value of MTU is 1500 on all the interface of Netscaler appliance any value configured more than 1500 on the interface will make the interface as jumbo enabled. In case of cluster backplane interface MTU value will be changed to 1514 by default, user has to change the backplane interface value to maximum mtu configured on any of the interface in cluster system plus 14 bytes more for backplane interface if Jumbo is enabled on any of the interface in a cluster system. Changing the backplane will bring back the MTU of backplane interface to default value of 1500. If a channel is configured as backplane then the same holds true for channel as well as member interfaces. In case of channel if member interfaces is configured as different mtu then the highest MTU configured MTU is treated as the LA MTU if MTU is not specified on LA explicitly. Low MTU interfaces in channel will be taken out of LA distribution list.

actualMtu

MTU of the channel. This is the maximum frame size that the channel can process.

vlan

Native VLAN of the channel.

mac

MAC address of the channel.

lmac

Specifies a MAC address for the LA channels configured in NetScaler virtual appliances (VPX). This MAC address is persistent after each reboot. If you don't specify this parameter, a MAC address is generated randomly for each LA channel. These MAC addresses changes after each reboot.

uptime

Duration for which the channel is UP. (Example: 3 hours 1 minute 1 second). This value is reset when the channel state changes to DOWN.

downTime

Duration for which the channel is DOWN. (Example: 3 hours 1 minute 1 second). This value is reset when the channel state changes to UP.

reqMedia

Requested media setting for this channel. Since there is no media associated with LA, the displayed values carry no significance.

reqSpeed

Requested speed setting for this channel. Since no media are associated with LA, this speed is used to determine the threshold for the slave interfaces. If the speed of the member interface is less than the requested speed, that interface is considered inactive.

reqDuplex

Requested duplex setting for this channel. Since no media are associated with LA, the displayed values carry no significance.

reqFlowcontrol

Requested flow control setting for this channel. Since no media are associated with LA, the displayed values carry no significance.

media

Requested media setting for this interface.

speed

Actual speed setting for this channel.

duplex

Actual duplex setting for this interface.

flowControl

Actual flow control setting for this channel.

connDistr

Connection distribution setting on this Channel.

macdistr

MAC distribution setting on this Channel.

Mode

The mode(AUTO/MANNUAL) for the LA channel.

haMonitor

HA monitoring enabled or disabled for this channel.

state

Enable or disable the LA channel.

autoneg

Requested auto negotiation setting for this channel. Since no media are associated with LA, this setting has no effect.

autonegResult

Actual auto negotiation setting for this channel.

tagged

VLAN tags setting on this channel.

tagall

The appliance adds a four-byte 802.1q tag to every packet sent on this channel. ON applies tags for all the VLANs that are bound to this channel. OFF, applies the tag for all VLANs other than the native VLAN.

trunk

This is deprecated by tagall

taggedAny

Channel setting to accept/drop all tagged packets.

taggedAutolearn

Dynaminc vlan membership on this channel.

hangDetect

Hang detect for this channel.

hangReset

Hang reset for this channel.

linkState

The current state of the link associated with the interface. For logical interfaces (LA), the state of the link is dependent on the state of the slave interfaces. For the link to be UP at least one of the slave interfaces needs to be UP.

intfState

Current state of the specified interface. The interface state set to UP only if the link state is UP and administrative state is ENABLED.

rxpackets

Number of bytes received by all the slave interfaces of the channel since the NetScaler appliance was started or the interface statistics were cleared.

rxbytes

Number of packets received by all member interfaces since the NetScaler appliance was started or the interface statistics were cleared.

rxerrors

Number of inbound packets dropped by the hardware of the slave interfaces since the NetScaler appliance was started or the interface statistics were cleared. Possible causes of dropped packets are CRC, length (undersize or oversize), and alignment errors.

rxdrops

Number of inbound packets dropped by the channel's slave interfaces. Commonly dropped packets are multicast frames, spanning tree BPDUs, packets destined to a MAC not owned by the NetScaler when L2 mode is disabled, or packets tagged for a VLAN that is not bound to the interface. In most healthy networks, this statistic increments at a steady rate regardless of traffic load. A sharp spike in dropped packets generally indicates an issue with connected L2 switches, such as a forwarding database overflow resulting in packets being broadcast on all ports.

txpackets

Number of packets transmitted by slave interfaces of a channel since the NetScaler appliance was started or the interface statistics were cleared.

txbytes

Number of bytes transmitted by slave interfaces of a channel since the NetScaler appliance was started or the interface statistics were cleared.

txerrors

Number of outbound packets dropped by the hardware of a channel's slave interfaces since the NetScaler appliance was started or the interface statistics were cleared. Possible causes of dropped packets are length (undersize or oversize) errors and lack of resources.

txdrops

Number of packets dropped in transmission by a channel's slave interfaces for one of the following reasons:

- (1) VLAN mismatch.
- (2) Oversized packets.

(3) Interface congestion.

(4) Loopback packets sent on non-loopback interface.

inDisc

Number of error-free inbound packets discarded by a channel's slave interfaces because of a lack of resources (for example, insufficient receive buffers).

outDisc

Number of error-free outbound packets discarded by a channel's slave interfaces because of a lack of resources. This statistic is not available on:

(1) 10G ports of NetScaler MPX 12500/12500/15500-10G platforms.

(2) 10G data ports on NetScaler MPX 17500/19500/21500 platforms.

fctls

Number of times flow control is performed on a channel's slave interfaces because of pause frames.

hangs

Number of hangs that occurred on the channel's slave interfaces.

stsStalls

Number of status stalls that occurred on the channel's slave interfaces.

txStalls

Number of Tx stalls happened that occurred on the channel's slave interfaces.

rxStalls

Number of Rx stalls that occurred on the channel's slave interfaces.

bdgMuted

Number of times a channel's slave interfaces stopped transmitting and receiving packets because of MAC moves between ports.

vmac

Virtual MAC of this channel.

vmac6

Virtual MAC for IPv6 on this interface.

ifAlias

The alias name for the interface.

reqThroughput

Minimum required throughput for an interface. Failover is triggered if the operating throughput of a Link Aggregation (LA) channel for which HAMON is ON falls below this value.

lrMinThroughput

Specifies the minimum throughput threshold (in Mbps) to be met by the active subchannel. Setting this parameter automatically divides an LACP channel into logical subchannels, with one subchannel active and the others in standby mode. When the maximum supported throughput of the active channel falls below the lrMinThroughput value, link failover occurs and a standby subchannel becomes active.

linkRedundancy

Link Redundancy for Cluster LAG.

throughput

Actual throughput for the interface.

bandwidthHigh

High threshold value for the bandwidth usage of the LA channel, in Mbps. The NetScaler appliance generates an SNMP trap message when the bandwidth usage of the LA channel is greater than or equal to the specified high threshold value.

bandwidthNormal

Normal threshold value for the bandwidth usage of the LA channel, in Mbps. When the bandwidth usage of the LA channel returns to less than or equal to the specified normal threshold after exceeding the high threshold, the NetScaler appliance generates an SNMP trap message to indicate that the bandwidth usage has returned to normal.

ifnum

The interfaces bound to link aggregate channel.

backplane

The cluster backplane status of the LA. If the status is enabled, the LA is part of the cluster backplane. By default, the backplane status is disabled.

clearTime

Time since the interface stats are cleared last time.

slavestate

State of the member interfaces.

slavemedia

Media type of the member interfaces.

slavespeed

Speed of the member interfaces.

slaveduplex

Duplex of the member interfaces.

slaveflowctl

Flowcontrol of the member interfaces.

slavetime

UP time of the member interfaces.

lACPMode

The LACP mode of the specified interface. The possible values are:

1. Active: A port in active mode generates LACP protocol messages on a regular basis, regardless of any need expressed by its partner to receive them.
2. Passive: A port in passive mode generally does not transmit LACP messages unless its partner is in the active mode; that is, it does not speak unless spoken to.
3. Disabled: Removes the interface from the LA channel. If this is only interface in the LA channel, the LA channel is also deleted.

lACPTimeout

Time to wait for the LACPDU. If a LACPDU is not received within this interval, the NetScaler marks the link partner port as DOWN. Possible values: Long and Short. Long lacptimeout is 90 sec and Short LACP timeout is 3 sec.

lacpActorPriority

LACP Actor Priority. A LACP port priority is configured on each port using LACP. LACP uses the port priority with the port number to form the port identifier. The port priority determines which ports should be put in standby mode when there is a hardware limitation that prevents all compatible ports from aggregating.

lacpActorPortno

LACP Actor port number. LACP uses the port priority with the port number to form the port identifier.

lacpPartnerState

LACP Partner State. Whether the port is in Active or Passive negotiating state.

lacpPartnerTimeout

The timeout value for the information reviewed in LACPDUs. It can have values as SHORT or LONG. The SHORT timeout is 3s and the LONG timeout is 90s.

lacpPartnerAggregation

The Aggregation flag indicates that the participant will allow the link to be used as part of an aggregate. Otherwise the link is to be used as an individual link, i.e. not aggregated with any other.

lacpPartnerInsync

The Synchronization flag indicates that the transmitting participant's mux component is in sync with the system id and key information transmitted.

lacpPartnerCollecting

The Collecting flag indicates that the participant's collector, i.e. the reception component of the mux, is definitely on. If set the flag communicates collecting.

lacpPartnerDistributing

The Distributing flag indicates that the participant's distributor is not definitely off. If reset the flag indicates not distributing.

lacpPartnerDefaulted

If the timer expires in the Expired state, the Receive Machine enters the Defaulted state.

lacpPartnerExpired

If the LACPDUs are received for timeout period, the Receive Machine enters the Expired state and the timer is restarted with the timeout value of SHORT timeout

lacpPartnerPriority

LACP Partner Priority. A LACP port priority is configured on each port using LACP. LACP uses the port priority with the port number to form the port identifier.

The port priority determines which ports should be put in standby mode when there is a hardware limitation that prevents all compatible ports from aggregating.

lacpPartnerSystemMac

LACP Partner System MAC.

lacpPartnerSystemPriority

LACP Partner System Priority. The LACP partner's system priority. The values for the priority range from 0 to 65535. The lower the value, the higher the system priority. The switch with the lower system priority value determines which links between LACP partner are active and which are in the standby for each LACP Channel.

lACPPartnerPortno

LACP Partner Port number. LACP uses the port priority with the port number to form the port identifier.

lACPPartnerKey

LACP Partner Key. The LACP key used by the partner port.

lACPActorAggregation

The Aggregation flag indicates that the participant will allow the link to be used as part of an aggregate. Otherwise the link is to be used as an individual link, i.e. not aggregated with any other.

lACPActorInsync

The Synchronization flag indicates that the transmitting participant's mux component is in sync with the system id and key information transmitted.

lACPActorCollecting

The Collecting flag indicates that the participant's collector, i.e. the reception component of the mux, is definitely on. If set the flag communicates collecting.

lACPActorDistributing

The Distributing flag indicates that the participant's distributor is not definitely off. If reset the flag indicates not distributing.

lACPPortMuxState

LACP Port MUX state. The state of the MUX control machine. The Mux Control Machine attaches the physical port to an aggregate port, using the Selection Logic to choose an appropriate port, and turns the distributor and collector for the physical port on or off as required by protocol information.

lACPPortRxStat

LACP Port RX state. The state of the Receive machine. The Receive Machine maintains partner information, recording protocol information from LACPDUs sent by remote partner(s). Received information is subject to a timeout, and if sufficient time elapses the receive machine will revert to using default partner information.

lACPPortSelectState

LACP Port SELECT state. The state of the SELECT state machine, It could be SELECTED or UNSELECTED.

lldpmode

Link Layer Discovery Protocol (LLDP) mode for an interface. The resultant LLDP mode of an interface depends on the LLDP mode configured at the global and the interface levels.

devno**count**

ci

The following operations can be performed on "ci":

show ci

Displays all the critical interfaces of the NetScaler appliance. In a High Availability configuration, an interface that has HA MON enabled and is not bound to any FIS, is a critical interface. Failure of any critical interface triggers HA failover.

Synopsys

show ci

Outputs

ifaces

Interfaces that are critical for the appliance to operate in high availability mode.

devno

count

stateflag

Example

```
>show ci Critical Interfaces: LO/1 1/2
```

fis

The following operations can be performed on "fis":

add | **rm** | **bind** | **unbind** | **show**

add fis

Adds a failover interface set (FIS) to the NetScaler appliance. A FIS is a logical group of interfaces. In an HA configuration, using a FIS is a way to prevent failover by grouping interfaces so that, when one interface fails, other functioning interfaces are still available. A FIS can also be configured for the nodes of a NetScaler cluster.

Synopsys

add fis <name> [-ownerNode <positive_integer>]

Arguments

name

Name for the FIS to be created. Leading character must be a number or letter. Other characters allowed, after the first character, are @ _ - . (period) : (colon) # and space (). Note: In a cluster setup, the FIS name on each node must be unique.

ownerNode

ID of the cluster node for which you are creating the FIS. Can be configured only through the cluster IP address.

Default value: -1

Minimum value: 0

Maximum value: 31

rm fis

Removes an FIS from the NetScaler appliance. When an FIS is removed, its interfaces are marked as critical interfaces.

Synopsys

rm fis <name>

Arguments

name

Name of the FIS that you want to remove from the NetScaler appliance.

bind fis

Binds the specified interfaces to a FIS.

Synopsys

bind fis <name> <ifnum> ...

Arguments

name

The name of the FIS to which you want to bind interfaces.

ifnum

Interface to be bound to the FIS, specified in slot/port notation (for example, 1/3).

unbind fis

Unbinds the specified interfaces from a FIS. An unbound interface becomes a critical interface if it is enabled and HA MON is on.

Synopsys

```
unbind fis <name> <ifnum> ...
```

Arguments

name

Name of the FIS from which to unbind interfaces.

ifnum

Interfaces to unbind from the FIS, specified in slot/port notation (for example, 1/3). Use spaces to separate multiple entries.

show fis

Displays the configured FISs.

Synopsys

```
show fis [<name>]
```

Arguments

name

The name of the FIS configured on the appliance.

Outputs

ifaces

Interfaces to be bound to the FIS, in slot/port notation (for example, 1/3).

stateflag

Used internally for display.

ifnum

Interface to be bound to the FIS, specified in slot/port notation (for example, 1/3)

ownerNode

ID of the cluster node for which you are creating the FIS. Can be configured only through the cluster IP address.

devno

count

Example

```
>show fis 1)      FIS: fis1      Member Interfaces : 1/1 Done
```

forwardingSession

The following operations can be performed on "forwardingSession":

[add](#) | [set](#) | [rm](#) | [show](#)

add forwardingSession

Adds a forwarding session rule, which creates forwarding-session entries for traffic that originates from or is destined for a particular network and is forwarded by the NetScaler appliance. By default, the appliance does not create session entries for traffic that only forwards (L3 mode). Add a forwarding session rule for a case in which a client request that the appliance forwards to a server results in a response that has to return by the same path

Synopsys

```
add forwardingSession <name> ((<network> [<netmask>]) | -acl6name <string> | -aclname <string>) [-td  
<positive_integer>] [-connfailover ( ENABLED | DISABLED )]
```

Arguments

name

Name for the forwarding session rule. Can begin with a letter, number, or the underscore character (_), and can consist of letters, numbers, and the hyphen (-), period (.), pound (#), space (), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after the rule is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my rule" or 'my rule').

network

An IPv4 network address or IPv6 prefix of a network from which the forwarded traffic originates or to which it is destined.

netmask

Subnet mask associated with the network.

acl6name

Name of any configured ACL6 whose action is ALLOW. The rule of the ACL6 is used as a forwarding session rule.

aclname

Name of any configured ACL whose action is ALLOW. The rule of the ACL is used as a forwarding session rule.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

connfailover

Synchronize connection information with the secondary appliance in a high availability (HA) pair. That is, synchronize all connection-related information for the forwarding session.

Possible values: ENABLED, DISABLED

Default value: DISABLED

set forwardingSession

Modifies parameters of a forwarding session rule.

Synopsys

```
set forwardingSession <name> [-connfailover ( ENABLED | DISABLED )]
```

Arguments

name

Name of the forwarding session rule. Required for identifying the forwarding session rule.

connfailover

Synchronize connection information with the secondary appliance in a high availability (HA) pair. That is, synchronize all connection-related information for the forwarding session.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
set forwardsession fw1 -connfailover enabled.
```

rm forwardingSession

Removes a forwarding session rule from the NetScaler appliance.

Synopsys

```
rm forwardingSession <name>
```

Arguments

name

Name of the forwarding session rule to be removed.

Example

```
rm forwardsession name.
```

show forwardingSession

Displays the settings of all forwarding session rules configured on the NetScaler appliance, or of the specified forwarding session rule.

Synopsys

```
show forwardingSession [<name>]
```

Arguments

name

Name of the forwarding session rule whose details you want to display.

Outputs

network

An IPv4 network address or IPv6 prefix of a network from which the forwarded traffic originates or to which it is destined.

netmask

Subnet mask associated with the network.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

aclname

Name of any configured ACL whose action is ALLOW. The rule of the ACL is used as a forwarding session rule.

acl6name

Name of any configured ACL6 whose action is ALLOW. The rule of the ACL6 is used as a forwarding session rule.

connfailover

Synchronize connection information with the secondary appliance in a high availability (HA) pair. That is, synchronize all connection-related information for the forwarding session.

devno

count

stateflag

inat

The following operations can be performed on "inat":

add | **rm** | **set** | **unset** | **stat** | **show**

add inat

Adds an INAT rule to the NetScaler appliance. When a packet generated by a client matches the conditions specified in the INAT rule, the appliance translates the packet's public destination IP address to a private destination IP address and forwards the packet to the server at that address.

Synopsys

```
add inat <name>@ <publicIP>@ <privateIP>@ [-tcpproxy ( ENABLED | DISABLED )] [-ftp ( ENABLED | DISABLED )] [-tftp ( ENABLED | DISABLED )] [-usip ( ON | OFF )] [-usnip ( ON | OFF )] [-proxyIP <ip_addr|ipv6_addr>] [-mode STATELESS] [-td <positive_integer>]
```

Arguments

name

Name for the Inbound NAT (INAT) entry. Leading character must be a number or letter. Other characters allowed, after the first character, are @ _ - . (period) : (colon) # and space ().

publicIP

Public IP address of packets received on the NetScaler appliance. Can be aNetScaler-owned VIP or VIP6 address.

privateIP

IP address of the server to which the packet is sent by the NetScaler. Can be an IPv4 or IPv6 address.

tcpproxy

Enable TCP proxy, which enables the NetScaler appliance to optimize the RNAT TCP traffic by using Layer 4 features.

Possible values: ENABLED, DISABLED

Default value: DISABLED

ftp

Enable the FTP protocol on the server for transferring files between the client and the server.

Possible values: ENABLED, DISABLED

Default value: DISABLED

tftp

To enable/disable TFTP (Default DISABLED).

Possible values: ENABLED, DISABLED

Default value: DISABLED

usip

Enable the NetScaler appliance to retain the source IP address of packets before sending the packets to the server.

Possible values: ON, OFF

Default value: OFF

usnip

Enable the NetScaler appliance to use a SNIP address as the source IP address of packets before sending the packets to the server.

Possible values: ON, OFF

Default value: ON

proxyIP

Unique IP address used as the source IP address in packets sent to the server. Must be a MIP or SNIP address.

mode

Stateless translation.

Possible values: STATELESS

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

Example

```
add nat mynat 1.2.3.4 192.168.1.100
```

rm inat

Remove the specified Inbound NAT configuration.

Synopsis

```
rm inat <name>@
```

Arguments

name

Name of the Inbound NAT entry to be removed from the NetScaler appliance.

Example

```
rm nat mynat.
```

set inat

Modifies parameters of an INAT rule.

Synopsis

```
set inat <name>@ [-privateIP <ip_addr|ipv6_addr>@] [-tcpproxy ( ENABLED | DISABLED )] [-ftp ( ENABLED | DISABLED )] [-tftp ( ENABLED | DISABLED )] [-usip ( ON | OFF )] [-usnip ( ON | OFF )] [-proxyIP <ip_addr|ipv6_addr>] [-mode STATELESS]
```

Arguments

name

The name of the Inbound NAT (INAT) entry that you want to modify.

privateIP

IP address of the server to which the packet is sent by the NetScaler. Can be an IPv4 or IPv6 address.

tcpproxy

Enable TCP proxy, which enables the NetScaler appliance to optimize the RNAT TCP traffic by using Layer 4 features.

Possible values: ENABLED, DISABLED

Default value: DISABLED

ftp

Enable the FTP protocol on the server for transferring files between the client and the server.

Possible values: ENABLED, DISABLED

Default value: DISABLED

tftp

To enable/disable TFTP (Default DISABLED).

Possible values: ENABLED, DISABLED

Default value: DISABLED

usip

Enable the NetScaler appliance to retain the source IP address of packets before sending the packets to the server.

Possible values: ON, OFF

Default value: OFF

usnip

Enable the NetScaler appliance to use a SNIP address as the source IP address of packets before sending the packets to the server.

Possible values: ON, OFF

Default value: ON

proxyIP

A unique IP address used as the source IP address in packets sent to the server. Must be a MIP or SNIP address.

mode

Stateless translation.

Possible values: STATELESS

Example

```
set nat mynat -tcpproxy ENABLED
```

unset inat

Use this command to remove inat settings. Refer to the set inat command for meanings of the arguments.

Synopsys

unset inat <name>@ [-tcpproxy] [-ftp] [-tftp] [-usip] [-usnip] [-proxyIP] [-mode]

stat inat

Display statistics for inat sessions.

Synopsys

stat inat [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

name

The INAT.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

TCP Packets translated (V4->V6) (nat46TotTcp46)

Total TCP packets translated (V4->v6).

UDP Packets translated (V4->V6) (nat46TotUdp46)

Total UDP packets translated (V4->v6).

ICMP Packets translated (V4->V6) (nat46TotIcmp46)

Total ICMP packets translated (V4->v6).

Total IPv4 packets dropped (nat46Totdrop46)

Total IPv4 packets dropped.

TCP Packets translated (V6->V4) (nat46TotTcp64)

Total TCP packets translated (V6->v4).

UDP Packets translated (V6->V4) (nat46TotUdp64)

Total UDP packets translated (V6->v4).

ICMP Packets translated (V6->V4) (nat46TotIcmp64)

Total ICMP packets translated (V6->v4).

Total IPv6 packets dropped (nat46Totdrop64)

Total IPv6 packets dropped.

TCP Packets translated (V4->V6) (inatNat46Tcp46)

TCP packets translated (V4->v6).

UDP Packets translated (V4->V6) (inatNat46Udp46)

UDP packets translated (V4->v6).

ICMP Packets translated (V4->V6) (inatNat46Icmp46)

ICMP packets translated (V4->v6).

IPv4 packets dropped (inatNat46drop46)

IPv4 packets dropped.

TCP Packets translated (V6->V4) (inatNat46Tcp64)

TCP packets translated (V6->v4).

UDP Packets translated (V6->V4) (inatNat46Udp64)

UDP packets translated (V6->v4).

ICMP Packets translated (V6->V4) (inatNat46Icmp64)

ICMP packets translated (V6->v4).

IPv6 packets dropped (inatNat46drop64)

IPv6 packets dropped.

Example

```
stat inat
```

show inat

show all configured inbound NAT.

Synopsys

```
show inat [<name>]
```

Arguments

name

Name for the Inbound NAT (INAT) entry. Leading character must be a number or letter. Other characters allowed, after the first character, are @ _ - . (period) : (colon) # and space ().

Outputs

publicIP

Public IP address of packets received on the NetScaler appliance. Can be a NetScaler-owned VIP or VIP6 address.

privateIP

IP address of the server to which the packet is sent by the NetScaler. Can be an IPv4 or IPv6 address.

proxyIP

Source IP address for connection to a server.

tcpproxy

Enable TCP proxy, which enables the NetScaler appliance to optimize the RNAT TCP traffic by using Layer 4 features.

ftp

Enable the FTP protocol on the server for transferring files between the client and the server.

tftp

To enable/disable TFTP (Default DISABLED).

usip

Enable the NetScaler appliance to retain the source IP address of packets before sending the packets to the server.

usnip

Enable the NetScaler appliance to use a SNIP address as the source IP address of packets before sending the packets to the server.

flags

Flags for different modes

mode

Stateless translation.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

devno

count

stateflag

Example

```
show nat
```


inatparam

The following operations can be performed on "inatparam":

[set](#) | [unset](#) | [show](#)

set inatparam

Set the inat parameter

Synopsys

```
set inatparam [-nat46v6Prefix <ipv6_addr|*> [-td <positive_integer>]] [-nat46IgnoreTOS ( YES | NO )] [-nat46ZeroChecksum ( ENABLED | DISABLED )] [-nat46v6Mtu <positive_integer>] [-nat46FragHeader ( ENABLED | DISABLED )]
```

Arguments

nat46v6Prefix

The prefix used for translating packets received from private IPv6 servers into IPv4 packets. This prefix has a length of 96 bits (128-32 = 96). The IPv6 servers embed the destination IP address of the IPv4 servers or hosts in the last 32 bits of the destination IP address field of the IPv6 packets. The first 96 bits of the destination IP address field are set as the IPv6 NAT prefix. IPv6 packets addressed to this prefix have to be routed to the NetScaler appliance to ensure that the IPv6-IPv4 translation is done by the appliance.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

nat46IgnoreTOS

Ignore TOS.

Possible values: YES, NO

Default value: NO

nat46ZeroChecksum

Calculate checksum for UDP packets with zero checksum

Possible values: ENABLED, DISABLED

Default value: ENABLED

nat46v6Mtu

MTU setting for the IPv6 side. If the incoming IPv4 packet greater than this, either fragment or send icmp need fragmentation error.

Default value: 1280

Minimum value: 1280

Maximum value: 9216

nat46FragHeader

When disabled, translator will not insert IPv6 fragmentation header for non fragmented IPv4 packets

Possible values: ENABLED, DISABLED

Default value: ENABLED

Example

```
set inat parameter -nat46ignoretos YES
```

unset inatparam

Unset the inat parameter. Refer to the set inatparam command for meanings of the arguments.

Synopsys

```
unset inatparam [-nat46v6Prefix [-td <positive_integer>]]
```

Example

```
unset inatparam -nat46v6Prefix -td 1
```

show inatparam

Show the inat parameters.

Synopsys

```
show inatparam [-td <positive_integer>]
```

Arguments

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

Outputs

nat46v6Prefix

The prefix used for translating packets received from private IPv6 servers into IPv4 packets. This prefix has a length of 96 bits ($128 - 32 = 96$). The IPv6 servers embed the destination IP address of the IPv4 servers or hosts in the last 32 bits of the destination IP address field of the IPv6 packets. The first 96 bits of the destination IP address field are set as the IPv6 NAT prefix. IPv6 packets addressed to this prefix have to be routed to the NetScaler appliance to ensure that the IPv6-IPv4 translation is done by the appliance.

nat46IgnoreTOS

Ignore TOS.

nat46ZeroChecksum

Calculate checksum for UDP packets with zero checksum

nat46v6Mtu

MTU setting for the IPv6 side. If the incoming IPv4 packet greater than this, either fragment or send icmp need fragmentation error.

nat46FragHeader

When disabled, translator will not insert IPv6 fragmentation header for non fragmented IPv4 packets

devno

count

stateflag

Example

```
show inat params
```

inatsession

The following operations can be performed on "inatsession":

stat inatsession

Display statistics for stateful inat sessions.

Synopsys

```
stat inatsession <name> [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (
basic | full )]
```

Arguments

name

INAT name

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

INAT total sessions (inatTotHits)

INAT total sessions

INAT Current sessions (inatCurSessions)

INAT current sessions

INAT total Received Bytes (inatTotReceiveBytes)

INAT total Received Bytes

INAT total Sent Bytes (inatTotSentBytes)

INAT total Sent Bytes

INAT total Packets Received (inatTotpktreceived)

INAT total Packets Received

INAT total Packets Sent (inatTotpktsent)

INAT total Packets Sent

Example

```
stat inatsession inat_1
```

interface

The following operations can be performed on "interface":

clear | **set** | **unset** | **enable** | **disable** | **reset** | **show** | **stat**

clear interface

Resets the statistical counters of the specified interface.

Synopsis

```
clear interface <id>@
```

Arguments

id

Interface number, in C/U format, where C can take one of the following values:

- * 0 - Indicates a management interface.
- * 1 - Indicates a 1 Gbps port.
- * 10 - Indicates a 10 Gbps port.
- * LA - Indicates a link aggregation port.
- * LO - Indicates a loop back port.

U is a unique integer for representing an interface in a particular port group.

set interface

Modifies the parameters of an interface.

Synopsis

```
set interface <id>@ [-speed <speed>] [-duplex <duplex>] [-flowControl <flowControl>] [-autoneg ( DISABLED |  
ENABLED )] [-haMonitor ( ON | OFF )] [-mtu <positive_integer>] [-tagall ( ON | OFF )] [-lacpMode <lacpMode>] [-  
lacpKey <positive_integer>] [-lagtype ( NODE | CLUSTER )] [-lacpPriority <positive_integer>] [-lacpTimeout ( LONG  
| SHORT )] [-ifAlias <string>] [-throughput <positive_integer>] [-linkRedundancy ( ON | OFF )] [-bandwidthHigh  
<positive_integer>] [-bandwidthNormal <positive_integer>]] [-lldpmode <lldpmode>]
```

Arguments

id

ID of the Interface whose parameters you want to modify.

For a NetScaler appliance, specify the interface in C/U notation (for example, 1/3).

For a cluster configuration, specify the interface in N/C/U notation (for example, 2/1/3).

where C can take one of the following values:

- * 0 - Indicates a management interface.
- * 1 - Indicates a 1 Gbps port.
- * 10 - Indicates a 10 Gbps port.

U is a unique integer for representing an interface in a particular port group.

N is the ID of the node to which an interface belongs in a cluster configuration.

Use spaces to separate multiple entries.

speed

Ethernet speed of the interface, in Mbps.

Notes:

* If you set the speed as AUTO, the NetScaler appliance attempts to auto-negotiate or auto-sense the link speed of the interface when it is UP. You must enable auto negotiation on the interface.

* If you set a speed other than AUTO, you must specify the same speed for the peer network device. Mismatched speed and duplex settings between the peer devices of a link lead to link errors, packet loss, and other errors.

Some interfaces do not support certain speeds. If you specify an unsupported speed, an error message appears.

Possible values: AUTO, 10, 100, 1000, 10000, 40000

Default value: AUTO

duplex

Duplex mode for the interface. If you set the duplex mode to AUTO, the NetScaler appliance attempts to auto-negotiate the duplex mode of the interface when it is UP. You must enable auto negotiation on the interface. If you set a duplex mode other than AUTO, you must specify the same duplex mode for the peer network device. Mismatched speed and duplex settings between the peer devices of a link lead to link errors, packet loss, and other errors.

Possible values: AUTO, HALF, FULL

Default value: AUTO

flowControl

802.3x flow control setting for the interface. The 802.3x specification does not define flow control for 10 Mbps and 100 Mbps speeds, but if a Gigabit Ethernet interface operates at those speeds, the flow control settings can be applied. The flow control setting that is finally applied to an interface depends on auto-negotiation. With the ON option, the peer negotiates the flow control, but the appliance then forces two-way flow control for the interface.

Possible values: OFF, RX, TX, RXTX

Default value: OFF

autoneg

Auto-negotiation state of the interface. With the ENABLED setting, the NetScaler appliance auto-negotiates the speed and duplex settings with the peer network device on the link. The NetScaler appliance auto-negotiates the settings of only those parameters (speed or duplex mode) for which the value is set as AUTO.

Possible values: DISABLED, ENABLED

Default value: NSA_DVC_AUTONEG_ON

haMonitor

In a High Availability (HA) configuration, monitor the interface for failure events. In an HA configuration, an interface that has HA MON enabled and is not bound to any Failover Interface Set (FIS), is a critical interface. Failure or disabling of any critical interface triggers HA failover.

Possible values: ON, OFF

Default value: ON

mtu

The maximum transmission unit (MTU) is the largest packet size, measured in bytes excluding 14 bytes ethernet header and 4 bytes crc, that can be transmitted and received by this interface. Default value of MTU is 1500 on all the interface of Netscaler appliance any value configured more than 1500 on the interface will

make the interface as jumbo enabled. In case of cluster backplane interface MTU value will be changed to 1514 by default, user has to change the backplane interface value to maximum mtu configured on any of the interface in cluster system plus 14 bytes more for backplane interface if Jumbo is enabled on any of the interface in a cluster system. Changing the backplane will bring back the MTU of backplane interface to default value of 1500. If a channel is configured as backplane then the same holds true for channel as well as member interfaces. In case of channel if member interfaces is configured as different mtu then the highest MTU configured MTU is treated as the LA MTU if MTU is not specified on LA explicitly. Low MTU interfaces in channel will be taken out of LA distribution list.

Default value: 1500

Minimum value: 1500

Maximum value: 9216

tagall

Add a four-byte 802.1q tag to every packet sent on this interface. The ON setting applies the tag for this interface's native VLAN. OFF applies the tag for all VLANs other than the native VLAN.

Possible values: ON, OFF

Default value: OFF

lacpMode

Bind the interface to a LA channel created by the Link Aggregation control protocol (LACP).

Available settings function as follows:

- * Active - The LA channel port of the NetScaler appliance generates LACPDU messages on a regular basis, regardless of any need expressed by its peer device to receive them.

- * Passive - The LA channel port of the NetScaler appliance does not transmit LACPDU messages unless the peer device port is in the active mode. That is, the port does not speak unless spoken to.

- * Disabled - Unbinds the interface from the LA channel. If this is the only interface in the LA channel, the LA channel is removed.

Possible values: DISABLED, ACTIVE, PASSIVE

Default value: DISABLED

lacpKey

Integer identifying the LACP LA channel to which the interface is to be bound.

For an LA channel of the NetScaler appliance, this digit specifies the variable x of an LA channel in LA/x notation, where x can range from 1 to 8. For example, if you specify 3 as the LACP key for an LA channel, the interface is bound to the LA channel LA/3.

For an LA channel of a cluster configuration, this digit specifies the variable y of a cluster LA channel in CLA/(y-4) notation, where y can range from 5 to 8. For example, if you specify 6 as the LACP key for a cluster LA channel, the interface is bound to the cluster LA channel CLA/2.

Minimum value: 1

Maximum value: 8

lagtype

Type of entity (NetScaler appliance or cluster configuration) for which to create the channel.

Possible values: NODE, CLUSTER

Default value: NODE

lacpPriority

LACP port priority, expressed as an integer. The lower the number, the higher the priority. The NetScaler appliance limits the number of interfaces in an LA channel to sixteen.

Default value: 32768

Minimum value: 1

Maximum value: 65535

lacpTimeout

Interval at which the NetScaler appliance sends LACPDU messages to the peer device on the LA channel.

Available settings function as follows:

LONG - 30 seconds.

SHORT - 1 second.

Possible values: LONG, SHORT

Default value: NSA_LACP_TIMEOUT_LONG

ifAlias

Alias name for the interface. Used only to enhance readability. To perform any operations, you have to specify the interface ID.

Default value: " "

throughput

Low threshold value for the throughput of the interface, in Mbps. In an HA configuration, failover is triggered if the interface has HA MON enabled and the throughput is below the specified the threshold.

Minimum value: 0

Maximum value: 160000

linkRedundancy

Link Redundancy for Cluster LAG.

Possible values: ON, OFF

Default value: OFF

bandwidthHigh

High threshold value for the bandwidth usage of the interface, in Mbps. The NetScaler appliance generates an SNMP trap message when the bandwidth usage of the interface is greater than or equal to the specified high threshold value.

Minimum value: 0

Maximum value: 160000

bandwidthNormal

Normal threshold value for the bandwidth usage of the interface, in Mbps. When the bandwidth usage of the interface becomes less than or equal to the specified normal threshold after exceeding the high threshold, the NetScaler appliance generates an SNMP trap message to indicate that the bandwidth usage has returned to normal.

Minimum value: 0

Maximum value: 160000

lldpMode

Link Layer Discovery Protocol (LLDP) mode for an interface. The resultant LLDP mode of an interface depends on the LLDP mode configured at the global and the interface levels.

Possible values: NONE, TRANSMITTER, RECEIVER, TRANSCEIVER

unset interface

Use this command to remove interface settings. Refer to the set interface command for meanings of the arguments.

Synopsys

```
unset interface <id>@ [-speed] [-duplex] [-flowControl] [-autoneg] [-haMonitor] [-mtu] [-tagall] [-lacpMode] [-lacpKey]
[-lacpPriority] [-lacpTimeout] [-ifAlias] [-throughput] [-linkRedundancy] [-bandwidthHigh] [-bandwidthNormal] [-
lldpmode]
```

enable interface

Enables the interface. If the link is active, it can transmit and receive packets. Note: To view the status of an interface, use the show interface command.

Synopsys

```
enable interface <id>@
```

Arguments

id

Interface number, in C/U format, where C can take one of the following values:

- * 0 - Indicates a management interface.
- * 1 - Indicates a 1 Gbps port.
- * 10 - Indicates a 10 Gbps port.
- * LA - Indicates a link aggregation port.
- * LO - Indicates a loop back port.

U is a unique integer for representing an interface in a particular port group.

disable interface

Disables the interface from transmitting and receiving packets. The link remains active and the peer network device is unaware that the interface has been disabled. In a High Availability configuration, an interface that has HA MON enabled and is not bound to any Failover Interface Set (FIS), is a critical interface. Disabling or failure of any critical interface triggers HA failover. Note: To view the status of an interface, use the show interface command.

Synopsys

```
disable interface <id>@
```

Arguments

id

Interface number, in C/U format, where C can take one of the following values:

- * 0 - Indicates a management interface.
- * 1 - Indicates a 1 Gbps port.
- * 10 - Indicates a 10 Gbps port.
- * LA - Indicates a link aggregation port.
- * LO - Indicates a loop back port.

U is a unique integer for representing an interface in a particular port group.

reset interface

Restarts the interface but leaves the administrative state ENABLED or DISABLED and configuration unchanged. The link pertaining to the interface is reestablished with the existing settings.

Synopsys

reset interface <id>@

Arguments

id

Interface number, in C/U format, where C can take one of the following values:

- * 0 - Indicates a management interface.
- * 1 - Indicates a 1 Gbps port.
- * 10 - Indicates a 10 Gbps port.
- * LA - Indicates a link aggregation port.
- * LO - Indicates a loop back port.

U is a unique integer for representing an interface in a particular port group.

show interface

Displays the settings of all interfaces or of the specified interface on the NetScaler appliance. To display the settings of all interfaces, run the command without any parameters. To display the settings of a particular interface, specify the ID of the interface.

Synopsys

show interface [<id>@] show interface stats - alias for 'stat interface'

Arguments

id

Interface number, in C/U format, where C can take one of the following values:

- * 0 - Indicates a management interface.
- * 1 - Indicates a 1 Gbps port.
- * 10 - Indicates a 10 Gbps port.
- * LA - Indicates a link aggregation port.
- * LO - Indicates a loop back port.

U is a unique integer for representing an interface in a particular port group.

Outputs

stateflag

deviceName

Name of the interface.

unit

Unit number for this interface, signifying the sequence number in which this interface is discovered on this Netscaler.

description

Display the type of interface, the speeds at which this interface can operate, and, if applicable, the type of SFP,.

flags

Flags for this interface. Used for communicating the device states.

mtu

The maximum transmission unit (MTU) is the largest packet size, measured in bytes excluding 14 bytes ethernet header and 4 bytes crc, that can be transmitted and received by this interface. Default value of MTU is 1500 on all the interface of Netscaler appliance any value configured more than 1500 on the interface will make the interface as jumbo enabled. In case of cluster backplane interface MTU value will be changed to 1514 by default, user has to change the backplane interface value to maximum mtu configured on any of the interface in cluster system plus 14 bytes more for backplane interface if Jumbo is enabled on any of the interface in a cluster system. Changing the backplane will bring back the MTU of backplane interface to default value of 1500. If a channel is configured as backplane then the same holds true for channel as well as member interfaces. In case of channel if member interfaces is configured as different mtu then the highest MTU configured MTU is treated as the LA MTU if MTU is not specified on LA explicitly. Low MTU interfaces in channel will be taken out of LA distribution list.

actualMtu

MTU for this interface (the largest frame that can transit this interface).

vlan

Native VLAN for this interface.

mac

MAC address for this interface.

uptime

Duration for which the interface has been UP (Example: 3 hours 1 minute 1 second). This value is reset when the interface state changes to DOWN..

downTime

Duration for which the interface has been DOWN. This value is reset when the interface state changes to UP. (Example: 3 hours 1 minute 1 second).

reqMedia

Requested media setting for this interface.

reqSpeed

Requested speed setting for this interface.

reqDuplex

Requested duplex setting for this interface.

reqFlowcontrol

Requested flow control setting for this interface.

media

Actual media setting for this interface.

speed

Actual speed setting for this interface.

duplex

Actual duplex setting for this interface.

flowControl

Actual flow control setting for this interface.

connDistr

Connection distribution setting on this interface.

macdistr

MAC distribution setting on this interface.

Mode

The mode(AUTO/MANNUAL) for the LA channel.

haMonitor

HA monitor enabled or disabled for this interface.

state

Link state of the interface (UP/DOWN).

autoneg

Interface autonegotiation enabled or disabled.

autonegResult

Actual auto-negotiation setting for this interface.

tagged

VLAN tags setting on this channel.

tagall

VLAN tagging behavior on this interface. With the ON setting,, packets are tagged with all the VLANs that are bound to this interface. With the OFF setting, packets are tagged with the native VLAN.

trunk

This argument is deprecated by tagall.

taggedAny

Interface setting to accept/drop all tagged packets.

taggedAutolearn

Dynamic VLAN membership autolearning enabled or disabled on this interface.

hangDetect

Hang detection enabled or disabled for this interface.

hangReset

Hang reset enabled or disabled for this interface.

linkState

The current state of the link associated with the interface. For logical interfaces (LA), the state of the link is dependent on the state of the slave interfaces. For the link to be UP at least one of the slave interfaces needs to be UP.

intfState

Current state of the specified interface. The interface state set to UP only if the link state is UP and administrative state is ENABLED.

rxpackets

Number of packets received by an interface since the NetScaler appliance was started or the interface statistics were cleared.

rxbytes

Number of bytes received by an interface since the NetScaler appliance was started or the interface statistics were cleared.

rxerrors

Number of inbound packets dropped by the hardware on a specified interface since the NetScaler appliance was started or the interface statistics were cleared. Packets can be dropped because of CRC, length (undersize or oversize), or alignment errors.

rxdrops

Number of inbound packets dropped by the specified interface. Commonly dropped packets are multicast frames, spanning tree BPDUs, packets destined to a MAC not owned by the NetScaler appliance when L2 mode is disabled, or packets tagged for a VLAN that is not bound to the interface. In most healthy networks, this statistic increments at a steady rate regardless of traffic load. A sharp spike in dropped packets generally indicates an issue with connected L2 switches, such as a forwarding database overflow resulting in packets being broadcast on all ports.

txpackets

Number of packets transmitted by an interface since the NetScaler appliance was started or the interface statistics were cleared.

txbytes

Number of bytes transmitted by an interface since the NetScaler appliance was started or the interface statistics were cleared.

txerrors

Number of outbound packets dropped by the hardware on a specified interface since the NetScaler appliance was started or the interface statistics were cleared. Packets can be dropped because of length (undersize or oversize) errors or a lack of resources. This statistic is available only for:

- (1) Loop back interface (LO) of all platforms.
- (2) All data ports on the NetScaler 12000 platform.
- (3) Management ports on the Netscaler MPX 15000 and 17000 platforms.

txdrops

Number of packets dropped in transmission by the specified interface for one of the following reasons.

- (1) VLAN mismatch.
- (2) Oversized packets.
- (3) Interface congestion.
- (4) Loopback packets sent on non loop back interface.

inDisc

Number of error-free inbound packets discarded by the specified interface because of a lack of resources (for example, insufficient receive buffers).

outDisc

Number of error-free outbound packets discarded by the specified interface because of a lack of resources. This statistic is not available on:

(1) 10G ports of NetScaler MPX 12500/12500/15500-10G platforms.

(2) 10G data ports on NetScaler MPX 17500/19500/21500 platforms.

fctIs

Number of times flow control is performed on the specified interface because of received pause frames.

hangs

Number of times the specified interface detected hangs in the transmit and receive paths since the NetScaler appliance was started or the interface statistics were cleared.

stsStalls

Number of times the status updates for a specified interface were stalled since the NetScaler appliance was started or the interface statistics were cleared. A status stall is detected when the status of the interface is not updated by the NIC hardware within 0.8 seconds of the last update.

txStalls

Number of times the interface stalled, when transmitting packets, since the NetScaler appliance was started or the interface statistics were cleared. Transmit (Tx) stalls are detected when a packet posted for transmission is not transmitted in 4 seconds.

rxStalls

Number of times the interface stalled, when receiving packets, since the NetScaler appliance was started or the interface statistics were cleared. Receive (Rx) stalls are detected when the following conditions are met:

(1)The link is up for more than 10 minutes.

(2)Packets are transmitted, but no packets are received for 16 seconds.

bdgMacMoved

Number of MAC moves between ports. A high rate of MAC moves typically indicates a bridge loop between two interfaces.

bdgMuted

Number of times the specified interface stopped transmitting and receiving packets because of MAC moves between ports.

vmac

Virtual MAC of this interface.

vmac6

Virtual MAC for IPv6 of this interface.

lACPMode

The LACP mode of the specified interface. The possible values are:

1. Active: A port in active mode generates LACP protocol messages on a regular basis, regardless of any need expressed by its partner to receive them.

2. Passive: A port in passive mode is generally not transmit LACP messages unless its partner is in the active mode; that is, it does not communicate to the other appliance unless other appliance communicates with this appliance.

lACPKey

Identifies the channel to which the interface is bound. The possible values are 1, 2, 3, and 4.

lACPPriority

LACP port priority, expressed as an integer. The lower the number, the higher the priority. The NetScaler appliance limits the number of interfaces in an LA channel to sixteen.

lacpTimeout

Time to wait for the LACPDU. If an LACPDU is not received within this interval, the NetScaler marks the link partner port as DOWN. Possible values; Long, Short. Long lacptimeout is 90 sec and Short LACP timeout is 3 sec.

lagtype

Type of entity (NetScaler appliance or cluster configuration) for which to create the channel.

ifAlias

Alias name for the interface. Used only to enhance readability. To perform any operations, you have to specify the interface ID.

reqThroughput

Minimum required throughput for an interface. Failover is triggered if the operating throughput of a Link Aggregation (LA) channel for which HAMON is ON falls below this value. The possible values are:

1. 1000Mbps for 1G interfaces.
2. 10000Mbps for 10G interfaces.
3. 160000Mbps for Link Aggregation channels.

throughput

Actual throughput for the interface.

linkRedundancy

Link Redundancy for Cluster LAG.

bandwidthHigh

High threshold value for the bandwidth usage of the interface, in Mbps. The NetScaler appliance generates an SNMP trap message when the bandwidth usage of the interface is greater than or equal to the specified high threshold value.

bandwidthNormal

Normal threshold value for the bandwidth usage of the interface, in Mbps. When the bandwidth usage of the interface becomes less than or equal to the specified normal threshold after exceeding the high threshold, the NetScaler appliance generates an SNMP trap message to indicate that the bandwidth usage has returned to normal.

backplane

The cluster backplane status of the interface. If the status is enabled, the interface is part of the cluster backplane. By default, the backplane status is disabled.

ifnum

Contains the LA Master, if the interface is part of LA channel.

clearTime

Time since the interface stats are cleared last time.

slavestate

State of the member interfaces.

slavemedia

Media type of the member interfaces.

slavespeed

Speed of the member interfaces.

slaveduplex

Duplex of the member interfaces.

slaveflowctl

Flowcontrol of the member interfaces.

slavetime

UP time of the member interfaces.

intftype

Interface Type, this field will have the interface type either it is virtual, physical or loopback.

lACPActorMode

* Active - The LA channel port of the NetScaler appliance generates LACPDU messages on a regular basis, regardless of any need expressed by its peer device to receive them.

* Passive - The LA channel port of the NetScaler appliance does not transmit LACPDU messages unless the peer device port is in the active mode. That is, the port does not speak unless spoken to.

* Disabled - Unbinds the interface from the LA channel. If this is the only interface in the LA channel, the LA channel is removed.

lACPActorTimeout

Interval at which the NetScaler appliance sends LACPDU messages to the peer device on the LA channel.

Available settings function as follows:

LONG - 30 seconds.

SHORT - 1 second.

lACPActorPriority

LACP Actor Priority. A LACP port priority is configured on each port using LACP. LACP uses the port priority with the port number to form the port identifier. The port priority determines which ports should be put in standby mode when there is a hardware limitation that prevents all compatible ports from aggregating.

lACPActorPortno

LACP Actor port number. LACP uses the port priority with the port number to form the port identifier.

lACPPartnerState

LACP Partner State. Whether the port is in Active or Passive negotiating state.

lACPPartnerTimeout

The timeout value for the information reviewed in LACPDUs. It can have values as SHORT or LONG. The SHORT timeout is 3s and the LONG timeout is 90s.

lACPPartnerAggregation

The Aggregation flag indicates that the participant will allow the link to be used as part of an aggregate. Otherwise the link is to be used as an individual link, i.e. not aggregated with any other.

lACPPartnerInsync

The Synchronization flag indicates that the transmitting participant's mux component is in sync with the system id and key information transmitted.

lACPPartnerCollecting

The Collecting flag indicates that the participant.s collector, i.e. the reception component of the mux, is definitely on. If set the flag communicates collecting.

lACPPartnerDistributing

The Distributing flag indicates that the participant.s distributor is not definitely off. If reset the flag indicates not distributing.

lACPPartnerDefaulted

If the timer expires in the Expired state, the Receive Machine enters the Defaulted state.

lACPPartnerExpired

If the LACPDUs are received for timeout period, the Receive Machine enters the Expired state and the timer is restarted with the timeout value of SHORT timeout

lACPPartnerPriority

LACP Partner Priority. A LACP port priority is configured on each port using LACP. LACP uses the port priority with the port number to form the port identifier.

The port priority determines which ports should be put in standby mode when there is a hardware limitation that prevents all compatible ports from aggregating.

lACPPartnerSystemMac

LACP Partner System MAC.

lACPPartnerSystemPriority

LACP Partner System Priority. The LACP partner's system priority. The values for the priority range from 0 to 65535. The lower the value, the higher the system priority. The switch with the lower system priority value determines which links between LACP partner are active and which are in the standby for each LACP Channel.

lACPPartnerPortno

LACP Partner Port number. LACP uses the port priority with the port number to form the port identifier.

lACPPartnerKey

LACP Partner Key. The LACP key used by the partner port.

lACPActorAggregation

The Aggregation flag indicates that the participant will allow the link to be used as part of an aggregate. Otherwise the link is to be used as an individual link, i.e. not aggregated with any other.

lACPActorInsync

The Synchronization flag indicates that the transmitting participant.s mux component is in sync with the system id and key information transmitted.

lACPActorCollecting

The Collecting flag indicates that the participant.s collector, i.e. the reception component of the mux, is definitely on. If set the flag communicates collecting.

lACPActorDistributing

The Distributing flag indicates that the participant.s distributor is not definitely off. If reset the flag indicates not distributing.

lACPPortMuxState

LACP Port MUX state. The state of the MUX control machine. The Mux Control Machine attaches the physical port to an aggregate port, using the Selection Logic to choose an appropriate port, and turns the distributor and collector for the physical port on or off as required by protocol information.

lACPPortRxStat

LACP Port RX state. The state of the Receive machine. The Receive Machine maintains partner information, recording protocol information from LACPDUs sent by remote partner(s). Received information is subject to a timeout, and if sufficient time elapses the receive machine will revert to using default partner information.

lACPPortSelectState

LACP Port SELECT state. The state of the SELECT state machine, It could be SELECTED or UNSELECTED.

lldpmode

Link Layer Discovery Protocol (LLDP) mode for an interface. The resultant LLDP mode of an interface depends on the LLDP mode configured at the global and the interface levels.

devno

count

Example

The output for the show interface command is as follows: 1) Interface 0/1 (Gig Ethe:

stat interface

Displays the statistics of all interfaces or of the specified interface on the NetScaler appliance. To display the statistics of all interfaces, run the command without any parameters. To display the statistics of a particular interface, specify the ID of the interface.

Synopsys

stat interface [<id>@] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

id

Interface number, in C/U format, where C can take one of the following values:

- * 0 - Indicates a management interface.
- * 1 - Indicates a 1 Gbps port.
- * 10 - Indicates a 10 Gbps port.
- * LA - Indicates a link aggregation port.
- * LO - Indicates a loop back port.

U is a unique integer for representing an interface in a particular port group.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Interface State (IntfState)

Current state of the specified interface. The interface state set to UP only if the link state is UP and administrative state is ENABLED .

Link uptime (UpTime)

Duration for which the link is UP. This statistic is reset when the state changes to DOWN.

Link downtime (DnTime)

Duration for which the link is DOWN. This statistic is reset when the state changes to UP.

Bytes received (Rx Bytes)

Number of bytes received by an interface since the NetScaler appliance was started or the interface statistics were cleared.

Bytes transmitted (Tx Bytes)

Number of bytes transmitted by an interface since the NetScaler appliance was started or the interface statistics were cleared.

Packets received (Rx Pkts)

Number of packets received by an interface since the NetScaler appliance was started or the interface statistics were cleared.

Packets transmitted (Tx Pkts)

Number of packets transmitted by an interface since the NetScaler appliance was started or the interface statistics were cleared.

Jumbo Packets Received (JumboRcv)

Number of Jumbo Packets received on this interface.

Jumbo Packets Transmitted (JumboXmit)

Number of Jumbo packets transmitted on this interface by upper layer, with TSO enabled actual transmission size could be non Jumbo.

Multicast packets (McastPkt)

Number of multicast packets received by the specified interface since the NetScaler appliance was started or the interface statistics were cleared.

NetScaler packets (NSPkt)

Number of packets, destined to the NetScaler, received by an interface since the NetScaler appliance was started or the interface statistics were cleared. The packets destined to NetScaler are those that have the same MAC address as that of an interface or a VMAC address owned by the NetScaler.

LACPDUs received (RxLacpdu)

Number of Link Aggregation Control Protocol Data Units(LACPDUs) received by the specified interface since the NetScaler appliance was started or the interface statistics were cleared.

LACPDUs transmitted (TxLacpdu)

Number of Link Aggregation Control Protocol Data Units(LACPDUs) transmitted by the specified interface since the NetScaler appliance was started or the interface statistics were cleared.

Error packets received (hw) (ErrRx)

Number of inbound packets dropped by the hardware on a specified interface once the NetScaler appliance starts or the interface statistics are cleared. This happens due to following reasons:

- 1) The hardware receives packets at a rate higher rate than that at which the software is processing packets. In this case, the hardware FIFO overruns and starts dropping the packets .
- 2) The specified interface fails to receive inbound packets from the appliance because of insufficient memory.
- 3) The specified interface receives packets with CRC errors (Alignment or Frame Check Sequence).
- 4) The specified interface receives overly long packets.
- 5) The specified interface receives packets with alignment errors.
- 6) The software does less buffering because it is running out of available memory. When hardware detects that there is no space into which to push newly arrived packets, it starts dropping them.
- 7) The specified interface receives packets with Frame Check Sequence (FCS) errors.
- 8) The specified interface receives packets smaller than 64 bytes.
- 9) The specified interface discards error-free inbound packets because of insufficient resources. For example: NIC buffers.
- 10) Packets are missed because of collision detection, link lost, physical decoding error, or MAC abort.

Error packets transmitted (hw) (ErrTx)

Number of outbound packets dropped by the hardware on a specified interface since the NetScaler appliance was started or the interface statistics were cleared. This could happen due to length (undersize or oversize) errors and lack of resources. This statistic is available only for:

- (1) Loop back interface (LO) of all platforms.
- (2) All data ports on the NetScaler 12000 platform.
- (3) Management ports on the MPX 15000 and 17000 platforms.

Inbound packets discarded(hw) (InDisc)

Number of error-free inbound packets discarded by the specified interface due to a lack of resources, for example, insufficient receive buffers.

Outbound packets discarded(hw) (OutDisc)

Number of error-free outbound packets discarded by the specified interface due to a lack of resources. This statistic is not available on:

- (1) 10G ports of NetScaler MPX 12500/12500/15500-10G platforms.
- (2) 10G data ports on NetScaler MPX 17500/19500/21500 platforms.

Packets dropped in Rx (sw) (DrpRxPkt)

Number of inbound packets dropped by the specified interface. Commonly dropped packets are multicast frames, spanning tree BPDUs, packets destined to a MAC not owned by the NetScaler when L2 mode is disabled, or packets tagged for a VLAN that is not bound to the interface. This statistic will increment in most healthy networks at a steady rate regardless of traffic load. If a sharp spike in dropped packets occurs, it generally indicates an issue with connected L2 switches, such as a forwarding database overflow resulting in packets being broadcast on all ports.

Packets dropped in Tx (sw) (DrpTxPkt)

Number of packets dropped in transmission by the specified interface due to one of the following reasons.

- (1) VLAN mismatch.
- (2) Oversized packets.
- (3) Interface congestion.
- (4) Loopback packets sent on non loop back interface.

NIC hangs (Hangs)

Number of times the specified interface detected hangs in the transmit and receive paths since the NetScaler appliance was started or the interface statistics were cleared.

Status stalls (StsStall)

Number of times the status updates for a specified interface were stalled since the NetScaler appliance was started or the interface statistics were cleared. A status stall is detected when the status of the interface is not updated by the NIC hardware within 0.8 seconds of the last update.

Transmit stalls (TxStall)

Number of times the interface stalled, when transmitting packets, since the NetScaler appliance was started or the interface statistics were cleared. Transmit (Tx) stalls are detected when a packet posted for transmission is not transmitted in 4 seconds.

Receive stalls (RxStall)

Number of times the interface stalled, when receiving packets, since the NetScaler appliance was started or the interface statistics were cleared. Receive (Rx) stalls are detected when the following conditions are met:

- (1) The link is up for more than 10 minutes.
- (2) Packets are transmitted, but no packets are received for 16 seconds.

Error-disables (ErrDis)

Number of times the specified interface is disabled by the NetScaler, due to continuous Receive (Rx) or Transmit (Tx) stalls, since the NetScaler appliance was started or the interface statistics were cleared. The NetScaler disables an interface when one of the following conditions is met:

- (1) Three consecutive transmit stalls occur with at most a gap of 10 seconds between any two stalls.
- (2) Three consecutive receive stalls occur with at most a gap of 120 seconds between any two stalls.

Duplex mismatches (DupMism)

Number of times duplex mismatches were detected on the specified interface since the NetScaler appliance was started or the interface statistics were cleared. A mismatch will occur if the duplex mode is not identically set on both ends of the link. This statistic is only available on the NetScaler Classic edition.

Link re-initializations (LnkReint)

Number of times the link has been re-initialized. A re-initialization occurs due to link state change, configuration parameter change, or administrative reset operation.

MAC moves registered (MacMvd)

Number of MAC moves between ports. If a high rate of MAC moves is observed, it is likely that there is a bridge loop between two interfaces.

Times NIC became muted (ErrMtd)

Number of times the specified interface stopped transmitting and receiving packets due to MAC moves between ports.

Interface Alias (IntfAlias)

Alias Name for the Interface

Link State (State)

The current state of the link associated with the interface. For logical interfaces (LA), the state of the link is dependent on the state of the slave interfaces. For the link to be UP at least one of the slave interfaces needs to be UP.

interfacePair

The following operations can be performed on "interfacePair":

[add](#) | [rm](#) | [show](#)

add interfacePair

Create an Interface Pair. Each Interface Pair or IFPAIR is identified by a IFID (integer from 1-255).

Synopsys

```
add interfacePair <id> -ifnum <interface_name> ...
```

Arguments

id

The Interface pair id

Minimum value: 1

Maximum value: 255

ifnum

The constituent interfaces in the interface pair

Minimum value: 1

rm interfacePair

Removes the IFPAIR created by the add intfPair command. Once the IFPAIR is removed, its interfaces become independent.

Synopsys

```
rm interfacePair <id>
```

Arguments

id

The Interface pair id

Minimum value: 1

Maximum value: 255

show interfacePair

Displays the configured Interface Pairs. If id is specified, then only that particular IFPAIR information is displayed. If it is not specified, all configured IFPAIRs are displayed.

Synopsys

```
show interfacePair [<id>]
```

Arguments

id

The Interface pair id

Minimum value: 1

Maximum value: 255

Outputs

ifnum

The constituent interfaces in the interface pair

ifaces

Names of all member interfaces of this Interface Pair

stateflag

state flag

devno

count

Example

An example of the output of the `show interfacepair` command is as follows: 1) IFPAIR

ip6Tunnel

The following operations can be performed on "ip6Tunnel":

[add](#) | [rm](#) | [show](#)

add ip6Tunnel

Creates an IPv6 tunnel. An IP tunnel is a communication channel, using encapsulation technologies, between two networks that do not have a routing path. Every IP packet that is shared between the two networks is encapsulated within another packet and then sent through the tunnel.

Synopsis

```
add ip6Tunnel <name> <remote> <local>
```

Arguments

name

Name for the IPv6 Tunnel. Cannot be changed after the service group is created. Must begin with a number or letter, and can consist of letters, numbers, and the @ _ - . (period) : (colon) # and space () characters.

remote

An IPv6 address of the remote NetScaler appliance used to set up the tunnel.

local

An IPv6 address of the local NetScaler appliance used to set up the tunnel.

Example

```
add ip6tunnel tun6 9901::200/64 *
```

rm ip6Tunnel

Removes an IPv6 tunnel from the NetScaler appliance.

Synopsis

```
rm ip6Tunnel <name>
```

Arguments

name

Name of the IPv6 tunnel to be removed.

Example

```
rm ip6tunnel tun6
```

show ip6Tunnel

Displays the settings of all IPv6 tunnels configured on the NetScaler appliance, or of the specified IPv6 tunnel.

Synopsis

```
show ip6Tunnel [<name> | <remote>]
```

Arguments

name

Name of the IPv6 tunnel whose details you want to display.

remote

The IPv6 address at which the remote NetScaler appliance connects to the tunnel.

Outputs

remoteIP

The remote IP address or subnet of the tunnel.

local

An IPv6 address of the local NetScaler appliance used to set up the tunnel.

type

The type of this tunnel.

encapIp

The effective local IP address of the tunnel. Used as the source of the encapsulated packets.

devno**count****stateflag**

Example

```
1) Name.....: tun61      Remote.....: 9901::200/64  Local.....:
```

ip6TunnelParam

The following operations can be performed on "ip6TunnelParam":

[set](#) | [unset](#) | [show](#)

set ip6TunnelParam

Sets global parameters of IPv6 tunnels on the NetScaler appliance.

Synopsys

```
set ip6TunnelParam [-srcIP <ipv6_addr|null>] [-dropFrag ( YES | NO )] [-dropFragCpuThreshold <positive_integer>]  
[-srcIPRoundRobin ( YES | NO )]
```

Arguments

srcIP

Common source IPv6 address for all IPv6 tunnels. Must be a SNIP6 or VIP6 address.

dropFrag

Drop any packet that requires fragmentation.

Possible values: YES, NO

Default value: NO

dropFragCpuThreshold

Threshold value, as a percentage of CPU usage, at which to drop packets that require fragmentation. Applies only if dropFragparameter is set to NO.

Minimum value: 1

Maximum value: 100

srcIPRoundRobin

Use a different source IPv6 address for each new session through a particular IPv6 tunnel, as determined by round robin selection of one of the SNIP6 addresses. This setting is ignored if a common global source IPv6 address has been specified for all the IPv6 tunnels. This setting does not apply to a tunnel for which a source IPv6 address has been specified.

Possible values: YES, NO

Default value: NO

Example

```
set ip6TunnelParam -srcIP 9901::100 -dropFrag YES -dropFragCpuThreshold 95
```

unset ip6TunnelParam

Resets the specified global parameters of IPv6 tunnels to their default settings. Refer to the set ip6TunnelParam command for parameter descriptions..Refer to the set ip6TunnelParam command for meanings of the arguments.

Synopsys

```
unset ip6TunnelParam [-srcIP] [-dropFrag] [-dropFragCpuThreshold] [-srcIPRoundRobin]
```

Example

```
unset ip6TunnelParam -srcIP -dropFrag -dropFragCpuThreshold
```

show ip6TunnelParam

Displays the global settings of IPv6 tunnels on the NetScaler appliance.

Synopsys

show ip6TunnelParam

Outputs

srcIP

Common source IPv6 address for all IPv6 tunnels. Must be a SNIP6 or VIP6 address.

dropFrag

Drop any packet that requires fragmentation.

dropFragCpuThreshold

Threshold value, as a percentage of CPU usage, at which to drop packets that require fragmentation. Applies only if dropFragparameter is set to NO.

srcIPRoundRobin

Use a different source IPv6 address for each new session through a particular IPv6 tunnel, as determined by round robin selection of one of the SNIP6 addresses. This setting is ignored if a common global source IPv6 address has been specified for all the IPv6 tunnels. This setting does not apply to a tunnel for which a source IPv6 address has been specified.

Example

```
Tunnel Source IP: 9901::100 Drop if Fragmentation Needed: YES CPU usage threshold to avo:
```

ipTunnel

The following operations can be performed on "ipTunnel":

[add](#) | [rm](#) | [show](#)

add ipTunnel

Creates an IPv4 tunnel. An IP tunnel is a communication channel, using encapsulation technologies, between two networks that do not have a routing path. Every IP packet that is shared between the two networks is encapsulated within another packet and then sent through the tunnel.

Synopsys

```
add ipTunnel <name> <remote> <remoteSubnetMask> <local> [-protocol <protocol> [-vlan <positive_integer>]] [-ipsecProfileName <string>]
```

Arguments

name

Name for the IP tunnel. Leading character must be a number or letter. Other characters allowed, after the first character, are @ _ - . (period) : (colon) # and space ().

remote

Public IPv4 address, of the remote device, used to set up the tunnel. For this parameter, you can alternatively specify a network address.

remoteSubnetMask

Subnet mask of the remote IP address of the tunnel.

local

Type of NetScaler owned public IPv4 address, configured on the local NetScaler appliance and used to set up the tunnel.

protocol

Name of the protocol to be used on this tunnel.

Possible values: IPIP, GRE, IPSEC, VXLAN

Default value: IPIP

ipsecProfileName

Name of IPSec profile to be associated.

Default value: "ns_ipsec_default_profile"

vlan

The vlan for mulicast packets

Minimum value: 1

Maximum value: 4094

Example

```
add iptunnel tunnel1 10.100.20.0 255.255.255.0 *
```

rm ipTunnel

Removes an IP tunnel configuration from the NetScaler appliance.

Synopsys

```
rm ipTunnel <name>
```

Arguments

name

Name of the IP Tunnel.

Example

```
rm iptunnel tunnell
```

show ipTunnel

Display the configured IP tunnels.

Synopsys

```
show ipTunnel [(<remote> <remoteSubnetMask>) | <name>]
```

Arguments

remote

Public IPv4 address, of the remote device, used to set up the tunnel. For this parameter, you can alternatively specify a network address.

remoteSubnetMask

Subnet mask of the remote IP address of the tunnel.

name

Name for the IP tunnel. Leading character must be a number or letter. Other characters allowed, after the first character, are @ _ - . (period) : (colon) # and space ().

Outputs

name

Name for the PBR

local

Type of NetScaler owned public IPv4 address, configured on the local NetScaler appliance and used to set up the tunnel.

protocol

Name of the protocol to be used on this tunnel.

type

The type of this tunnel.

encapIp

The effective local IP address of the tunnel. Used as the source of the encapsulated packets.

channel

The tunnel that is bound to a netbridge.

ipsecProfileName

Name of IPSec profile to be associated.

vlan

The vlan for mulicast packets

tunnelType

Indicates that a tunnel is User-Configured, Internal or DELETE-IN-PROGRESS.

ipsecTunnelStatus

Whether the ipsec on this tunnel is up or down.

devno**count****stateflag**

Example

```
1) Name.....: t1      Remote.....: 10.102.33.0  Mask.....: 255.255.255.0
```


ipTunnelParam

The following operations can be performed on "ipTunnelParam":

[set](#) | [unset](#) | [show](#)

set ipTunnelParam

Sets global parameters of IPv4 tunnels on the NetScaler appliance.

Synopsys

```
set ipTunnelParam [-srcIP <ip_addr>] [-dropFrag ( YES | NO )] [-dropFragCpuThreshold <positive_integer>] [-srcIPRoundRobin ( YES | NO )] [-enableStrictRx ( YES | NO )] [-enableStrictTx ( YES | NO )]
```

Arguments

srcIP

Common source-IP address for all tunnels. For a specific tunnel, this global setting is overridden if you have specified another source IP address. Must be a MIP or SNIP address.

dropFrag

Drop any IP packet that requires fragmentation before it is sent through the tunnel.

Possible values: YES, NO

Default value: NO

dropFragCpuThreshold

Threshold value, as a percentage of CPU usage, at which to drop packets that require fragmentation to use the IP tunnel. Applies only if dropFragparameter is set to NO. The default value, 0, specifies that this parameter is not set.

Minimum value: 1

Maximum value: 100

srcIPRoundRobin

Use a different source IP address for each new session through a particular IP tunnel, as determined by round robin selection of one of the SNIP addresses. This setting is ignored if a common global source IP address has been specified for all the IP tunnels. This setting does not apply to a tunnel for which a source IP address has been specified.

Possible values: YES, NO

Default value: NO

enableStrictRx

Strict PBR check for IPSec packets received through tunnel

Possible values: YES, NO

Default value: NO

enableStrictTx

Strict PBR check for packets to be sent IPSec protected

Possible values: YES, NO

Default value: NO

Example

```
set ipTunnelParam -srcIP 10.100.20.48 -dropFrag YES -dropFragCpuThreshold 95
```

unset ipTunnelParam

Use this command to remove ipTunnelParam settings. Refer to the set ipTunnelParam command for meanings of the arguments.

Synopsys

```
unset ipTunnelParam [-srcIP] [-dropFrag] [-dropFragCpuThreshold] [-srcIPRoundRobin] [-enableStrictRx] [-enableStrictTx]
```

show ipTunnelParam

Display the IP Tunnel global settings on the NetScaler

Synopsys

```
show ipTunnelParam
```

Outputs

srcIP

Common source-IP address for all tunnels. For a specific tunnel, this global setting is overridden if you have specified another source IP address. Must be a MIP or SNIP address.

dropFrag

Drop any IP packet that requires fragmentation before it is sent through the tunnel.

dropFragCpuThreshold

Threshold value, as a percentage of CPU usage, at which to drop packets that require fragmentation to use the IP tunnel. Applies only if dropFragparameter is set to NO. The default value, 0, specifies that this parameter is not set.

srcIPRoundRobin

Use a different source IP address for each new session through a particular IP tunnel, as determined by round robin selection of one of the SNIP addresses. This setting is ignored if a common global source IP address has been specified for all the IP tunnels. This setting does not apply to a tunnel for which a source IP address has been specified.

enableStrictRx

Strict PBR check for IPSec packets received through tunnel

enableStrictTx

Strict PBR check for packets to be sent IPSec protected

Example

```
Tunnel Source IP: 10.100.20.48 Drop if Fragmentation Needed: YES CPU usage threshold to %
```

ipset

The following operations can be performed on "ipset":

add | **rm** | **bind** | **unbind** | **show**

add ipset

Creates an IP set to which you can bind subnet IP (SNIP) or mapped IP (MIP) addresses that have been configured on the NetScaler appliance.

Synopsys

```
add ipset <name> [-td <positive_integer>]
```

Arguments

name

Name for the IP set. Must begin with a letter, number, or the underscore character (_), and can consist of letters, numbers, and the hyphen (-), period (.), pound (#), space (), at sign (@), equals (=), colon (:), and underscore characters. Cannot be changed after the IP set is created. Choose a name that helps identify the IP set.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

Example

```
add ipset pool1
```

rm ipset

Removes an IP set from the NetScaler appliance.

Synopsys

```
rm ipset <name> ...
```

Arguments

name

Name of the IP set to be removed.

Example

```
rm ipset pool1
```

bind ipset

Binds specified IP addresses to an IP set.

Synopsys

```
bind ipset <name> <IPAddress>@ ...
```

Arguments

name

Name of the IP set to which to bind IP addresses.

IPAddress

SNIP or MIP addresses, configured on the NetScaler appliance, to be bound to the IP set. (If using the CLI, use spaces to separate multiple addresses.)

Example

```
bind ipset ipset_1 10.102.1.10
```

unbind ipset

Unbinds the associated IP addresses from an IP set.

Synopsys

```
unbind ipset <name> <IPAddress>@ ...
```

Arguments

name

Name of the IP set from which to unbind IP addresses.

IPAddress

IP addresses to be unbound from the IP set. (If using the CLI, use spaces to separate multiple addresses.)

Example

```
unbind ipset ipset_1 10.102.1.10
```

show ipset

Displays the settings of all IP sets configured on the NetScaler appliance, or of the specified IP set.

Synopsys

```
show ipset [<name>]
```

Arguments

name

Name of the IP set whose details you want to display.

Outputs

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

IPAddress

One or more IP addresses bound to the IP set.

stateflag

state flag

flags

ipSetRefCount

Used to keep reference count of IP

devno

count

Example

```
show network ipset
```

ipv6

The following operations can be performed on "ipv6":

[set](#) | [unset](#) | [show](#)

set ipv6

Sets the IPv6-related parameters.

Synopsys

```
set ipv6 [-rlearning ( ENABLED | DISABLED )] [-routerRedirection ( ENABLED | DISABLED )] [-ndBasereachTime <positive_integer>] [-ndRetransmissionTime <positive_integer>] [-natprefix <ipv6_addr|*> [-td <positive_integer>]] [-doDAD ( ENABLED | DISABLED )]
```

Arguments

rlearning

Enable the NetScaler appliance to learn about various routes from Router Advertisement (RA) and Router Solicitation (RS) messages sent by the routers.

Possible values: ENABLED, DISABLED

Default value: DISABLED

routerRedirection

Enable the NetScaler appliance to do Router Redirection.

Possible values: ENABLED, DISABLED

Default value: DISABLED

ndBasereachTime

Base reachable time of the Neighbor Discovery (ND6) protocol. The time, in milliseconds, that the NetScaler appliance assumes an adjacent device is reachable after receiving a reachability confirmation.

Default value: 30000

Minimum value: 1

ndRetransmissionTime

Retransmission time of the Neighbor Discovery (ND6) protocol. The time, in milliseconds, between retransmitted Neighbor Solicitation (NS) messages, to an adjacent device.

Default value: 1000

Minimum value: 1

natprefix

Prefix used for translating packets from private IPv6 servers to IPv4 packets. This prefix has a length of 96 bits ($128 - 32 = 96$). The IPv6 servers embed the destination IP address of the IPv4 servers or hosts in the last 32 bits of the destination IP address field of the IPv6 packets. The first 96 bits of the destination IP address field are set as the IPv6 NAT prefix. IPv6 packets addressed to this prefix have to be routed to the NetScaler appliance to ensure that the IPv6-IPv4 translation is done by the appliance.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

doDAD

Enable the NetScaler appliance to do Duplicate Address Detection (DAD) for all the NetScaler owned IPv6 addresses regardless of whether they are obtained through stateless auto configuration, DHCPv6, or manual configuration.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
set ipv6 -natprefix 2000::/96
```

unset ipv6

Unset the IPv6-related parameters: RA Learning and IPv6 NAT Prefix..Refer to the set ipv6 command for meanings of the arguments.

Synopsys

```
unset ipv6 [-rlearning] [-routerRedirection] [-ndBasereachTime] [-ndRetransmissionTime] [-natprefix [-td  
<positive_integer>]] [-doDAD]
```

Example

```
unset ipv6 -natprefix -td 1
```

show ipv6

Display IPv6 settings

Synopsys

```
show ipv6 [-td <positive_integer>]
```

Arguments

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

Outputs

rlearning

Enable the NetScaler appliance to learn about various routes from Router Advertisement (RA) and Router Solicitation (RS) messages sent by the routers.

routerRedirection

Enable the NetScaler appliance to do Router Redirection.

basereachtime

ND6 base reachable time (ms)

ndBasereachTime

Base reachable time of the Neighbor Discovery (ND6) protocol. The time, in milliseconds, that the NetScaler appliance assumes an adjacent device is reachable after receiving a reachability confirmation.

reachtime

ND6 computed reachable time (ms)

ndreachtime

ND6 computed reachable time (ms)

retransmissiontime

ND6 retransmission time (ms)

ndRetransmissionTime

Retransmission time of the Neighbor Discovery (ND6) protocol. The time, in milliseconds, between retransmitted Neighbor Solicitation (NS) messages, to an adjacent device.

natprefix

Prefix used for translating packets from private IPv6 servers to IPv4 packets. This prefix has a length of 96 bits ($128 - 32 = 96$). The IPv6 servers embed the destination IP address of the IPv4 servers or hosts in the last 32 bits of the destination IP address field of the IPv6 packets. The first 96 bits of the destination IP address field are set as the IPv6 NAT prefix. IPv6 packets addressed to this prefix have to be routed to the NetScaler appliance to ensure that the IPv6-IPv4 translation is done by the appliance.

doDAD

Enable the NetScaler appliance to do Duplicate Address Detection (DAD) for all the NetScaler owned IPv6 addresses regardless of whether they are obtained through stateless auto configuration, DHCPv6, or manual configuration.

devno

count

stateflag

Example

```
show ipv6
```


lacp

The following operations can be performed on "lacp":

[set](#) | [show](#)

set lacp

Sets the Link Aggregation Control Protocol (LACP) system priority. Note: The NetScaler appliance automatically adds a parameter called mac in the configuration file (ns.conf) for this command entry. This parameter is set to the MAC address of one of the NetScaler appliance's interfaces and is used along with the system priority to form the system ID for the LACP channel.

Synopsys

```
set lacp -sysPriority <positive_integer> [-ownerNode <positive_integer>]
```

Arguments

sysPriority

Priority number that determines which peer device of an LACP LA channel can have control over the LA channel. This parameter is globally applied to all LACP channels on the NetScaler appliance. The lower the number, the higher the priority.

Default value: 32768

Minimum value: 1

Maximum value: 65535

ownerNode

The owner node in a cluster for which we want to set the lacp priority. Owner node can vary from 0 to 31. Ownernode value of 254 is used for Cluster.

Default value: 255

Minimum value: 0

show lacp

Displays the settings of all channels created by the link aggregation control protocol (LACP) on the NetScaler appliance.

Synopsys

```
show lacp [-ownerNode <positive_integer>]
```

Arguments

ownerNode

The owner node in a cluster for which we want to set the lacp priority. Owner node can vary from 0 to 31. Ownernode value of 254 is used for Cluster.

Default value: 255

Minimum value: 0

Outputs

deviceName

Name of the channel.

sysPriority

Priority number that determines which peer device of an LACP LA channel can have control over the LA channel. This parameter is globally applied to all LACP channels on the NetScaler appliance. The lower the number, the higher the priority.

mac

LACP system MAC.

flags

Flags of this channel.

lacpKey

LACP key of this channel.

clustersysPriority

LACP system (Cluster) priority

clusterMac

LACP system (Cluster) mac.

devno**count****stateflag**

linkset

The following operations can be performed on "linkset":

`add` | `rm` | `bind` | `unbind` | `show`

add linkset

Adds a linkset to the NetScaler cluster.

Synopsys

```
add linkset <id>
```

Arguments

id

Unique identifier for the linkset. Must be of the form LS/x, where x can be an integer from 1 to 32.

Example

```
add linkset LS/1
```

rm linkset

Removes a linkset from the cluster.

Synopsys

```
rm linkset <id>
```

Arguments

id

ID of the linkset to be removed.

Example

```
rm linkset LS/1
```

bind linkset

Binds interfaces to the linkset.

Synopsys

```
bind linkset <id> -ifnum <interface_name> ...
```

Arguments

id

ID of the linkset to which to bind the interfaces.

ifnum

The interfaces to be bound to the linkset.

Example

```
bind linkset LS/1 -ifnum 1/1/1
```

unbind linkset

Unbinds interfaces from the linkset.

Synopsys

```
unbind linkset <id> -ifnum <interface_name> ...
```

Arguments

id

ID of the linkset from which to unbind the interfaces.

ifnum

Interfaces to be unbound from the linkset.

Example

```
unbind linkset LS/1 -ifnum 1/1/1
```

show linkset

Displays information about all linksets, or displays information about the specified linkset.

Synopsys

```
show linkset [<id>]
```

Arguments

id

ID of the linkset for which to display information. If an ID is not provided, the display includes information about all linksets that are available in the cluster.

Outputs

ifnum

The interfaces to be bound to the linkset.

stateflag

state flag

devno

count

Example

```
show linkset
```

nat64

The following operations can be performed on "nat64":

add | **set** | **unset** | **rm** | **stat** | **show**

add nat64

Configure a nat64 rule on the appliance.

Synopsys

```
add nat64 <name> <acl6name> [-netProfile <string>]
```

Arguments

name

Name for the NAT64 rule. Must begin with a letter, number, or the underscore character (_), and can consist of letters, numbers, and the hyphen (-), period (.) pound (#), space (), at sign (@), equals (=), colon (:), and underscore characters. Cannot be changed after the rule is created. Choose a name that helps identify the NAT64 rule.

acl6name

Name of any configured ACL6 whose action is ALLOW. IPv6 Packets matching the condition of this ACL6 rule and destination IP address of these packets matching the NAT64 IPv6 prefix are considered for NAT64 translation.

netProfile

Name of the configured netprofile. The NetScaler appliance selects one of the IP address in the netprofile as the source IP address of the translated IPv4 packet to be sent to the IPv4 server.

set nat64

Set the configured nat64 rule.

Synopsys

```
set nat64 <name> [-acl6name <string>] [-netProfile <string>]
```

Arguments

name

Name for the NAT64 rule. Must begin with a letter, number, or the underscore character (_), and can consist of letters, numbers, and the hyphen (-), period (.) pound (#), space (), at sign (@), equals (=), colon (:), and underscore characters. Cannot be changed after the rule is created. Choose a name that helps identify the NAT64 rule.

acl6name

Name of any configured ACL6 whose action is ALLOW. IPv6 Packets matching the condition of this ACL6 rule and destination IP address of these packets matching the NAT64 IPv6 prefix are considered for NAT64 translation.

netProfile

Name of the configured netprofile. The NetScaler appliance selects one of the IP address in the netprofile as the source IP address of the translated IPv4 packet to be sent to the IPv4 server.

Example

```
set nat64 rule1 -acl6name acl1 .
```

unset nat64

Use this command to remove nat64 settings. Refer to the set nat64 command for meanings of the arguments.

Synopsys

```
unset nat64 <name> -netProfile
```

rm nat64

Remove the configured nat64 rule.

Synopsys

```
rm nat64 <name>
```

Arguments

name

Name for the NAT64 rule. Must begin with a letter, number, or the underscore character (_), and can consist of letters, numbers, and the hyphen (-), period (.) pound (#), space (), at sign (@), equals (=), colon (:), and underscore characters. Cannot be changed after the rule is created. Choose a name that helps identify the NAT64 rule.

Example

```
rm nat64 name.
```

stat nat64

Display statistics for nat64 sessions.

Synopsys

```
stat nat64 [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

TCP Sessions (nat64TotTcpSessions)

Total number of TCP sessions created by NAT64.

UDP Sessions (nat64TotUdpSessions)

Total number of UDP sessions created by NAT64.

ICMP Sessions (nat64TotIcmpSessions)

Total number of ICMP sessions created by NAT64.

Total Sessions (nat64TotSessions)

Total number of sessions created by NAT64.

Example

```
stat nat64
```

show nat64

Display the nat64 configuration.

Synopsys

```
show nat64 [<name>]
```

Arguments

name

Name for the NAT64 rule. Must begin with a letter, number, or the underscore character (_), and can consist of letters, numbers, and the hyphen (-), period (.), pound (#), space (), at sign (@), equals (=), colon (:), and underscore characters. Cannot be changed after the rule is created. Choose a name that helps identify the NAT64 rule.

Outputs

acl6name

Name of any configured ACL6 whose action is ALLOW. IPv6 Packets matching the condition of this ACL6 rule and destination IP address of these packets matching the NAT64 IPv6 prefix are considered for NAT64 translation.

netProfile

Name of the configured netprofile. The NetScaler appliance selects one of the IP address in the netprofile as the source IP address of the translated IPv4 packet to be sent to the IPv4 server.

devno

count

stateflag

nd6

The following operations can be performed on "nd6":

[add](#) | [clear](#) | [rm](#) | [show](#)

add nd6

Adds a static entry to the ND6 table of the NetScaler appliance.

Synopsys

```
add nd6 <neighbor> <mac> (<ifnum> | (-vxlan <positive_integer> -vtep <ip_addr>)) [-vlan <integer>] [-td <positive_integer>]
```

Arguments

neighbor

Link-local IPv6 address of the adjacent network device to add to the ND6 table.

mac

MAC address of the adjacent network device.

ifnum

Interface through which the adjacent network device is available, specified in slot/port notation (for example, 1/3). Use spaces to separate multiple entries.

vlan

Integer value that uniquely identifies the VLAN on which the adjacent network device exists.

Minimum value: 1

Maximum value: 4094

vxlan

ID of the VXLAN on which the IPv6 address of this ND6 entry is reachable.

Minimum value: 1

Maximum value: 16777215

vtep

IP address of the VXLAN tunnel endpoint (VTEP) through which the IPv6 address of this ND6 entry is reachable.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

Example

```
add nd6 2001::1 00:04:23:be:3c:06 5 1/1
```

clear nd6

Removes all IPv6 neighbour discovery entries from the NetScaler appliance.

Synopsys

clear nd6

rm nd6

Remove a static IPv6 neighbor discovery entry from the NetScaler appliance's ND6 table.

Synopsys

rm nd6 <neighbor> [-vlan <integer> | -vxlan <positive_integer>] [-td <positive_integer>]

Arguments

neighbor

Link-local IPv6 address of the adjacent network device that you want to remove from the ND6 table.

vlan

Integer value that uniquely identifies the VLAN for the ND6 entry you want to remove.

Minimum value: 1

Maximum value: 4094

vxlan

Integer value that uniquely identifies the VXLAN for the ND6 entry you want to remove.

Minimum value: 1

Maximum value: 16777215

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

Example

```
rm nd6 2001::1 5 1/1
```

show nd6

Display the neighbor discovery information.

Synopsys

show nd6 [<neighbor> [-td <positive_integer>]]

Arguments

neighbor

Link-local IPv6 address of the adjacent network device to add to the ND6 table.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

Outputs

mac

MAC address of the adjacent network device.

state

ND6 state

timeout

Time elapsed

ifnum

Interface through which the adjacent network device is available, specified in slot/port notation (for example, 1/3). Use spaces to separate multiple entries.

vlan

Integer value that uniquely identifies the VLAN on which the adjacent network device exists.

vxlan

ID of the VXLAN on which the IPv6 address of this ND6 entry is reachable.

vtep

IP address of the VXLAN tunnel endpoint (VTEP) through which the IPv6 address of this ND6 entry is reachable.

flags

flag for static/permanent entry.

channel

The tunnel that is bound to a netbridge.

devno

count

stateflag

Example

Following is an example of the output for the `show nd6` command:

Neighbor	MAC	i
----------	-----	---

nd6RAvariables

The following operations can be performed on "nd6RAvariables":

set | **unset** | **show** | **bind** | **unbind**

set nd6RAvariables

Set vlan specific Router Advertisement parameters in NetScaler.

Synopsys

```
set nd6RAvariables -vlan <positive_integer> [-ceaseRouterAdv ( YES | NO )] [-sendRouterAdv ( YES | NO )] [-srcLinkLayerAddrOption ( YES | NO )] [-onlyUnicastRtAdvResponse ( YES | NO )] [-managedAddrConfig ( YES | NO )] [-otherAddrConfig ( YES | NO )] [-currHopLimit <positive_integer>] [-maxRtAdvInterval <positive_integer>] [-minRtAdvInterval <positive_integer>] [-linkMTU <positive_integer>] [-reachableTime <positive_integer>] [-retransTime <positive_integer>] [-defaultLifeTime <integer>]
```

Arguments

vlan

The VLAN number.

Minimum value: 0

Maximum value: 4094

ceaseRouterAdv

Cease router advertisements on this vlan.

Possible values: YES, NO

Default value: NO

sendRouterAdv

whether the router sends periodic RAs and responds to Router Solicitations.

Possible values: YES, NO

Default value: NO

srcLinkLayerAddrOption

Include source link layer address option in RA messages.

Possible values: YES, NO

Default value: YES

onlyUnicastRtAdvResponse

Send only Unicast Router Advertisements in respond to Router Solicitations.

Possible values: YES, NO

Default value: NO

managedAddrConfig

Value to be placed in the Managed address configuration flag field.

Possible values: YES, NO

Default value: NO

otherAddrConfig

Value to be placed in the Other configuration flag field.

Possible values: YES, NO

Default value: NO

currHopLimit

Current Hop limit.

Default value: 64

Minimum value: 0

Maximum value: 255

maxRtAdvInterval

Maximum time allowed between unsolicited multicast RAs, in seconds.

Default value: 600

Minimum value: 4

Maximum value: 1800

minRtAdvInterval

Minimum time interval between RA messages, in seconds.

Default value: 198

Minimum value: 3

Maximum value: 1350

linkMTU

The Link MTU.

Default value: 0

Minimum value: 0

Maximum value: 1500

reachableTime

Reachable time, in milliseconds.

Default value: 0

Minimum value: 0

Maximum value: 3600000

retransTime

Retransmission time, in milliseconds.

Default value: 0

Minimum value: 0

defaultLifeTime

Default life time, in seconds.

Default value: 1800

Minimum value: 0

Maximum value: 9000

Example

```
set nd6RAvariables -vlan 2 -maxRtAdvInterval 600
```

unset nd6RAvariables

Use this command to remove nd6RAvariables settings. Refer to the set nd6RAvariables command for meanings of the arguments.

Synopsys

```
unset nd6RAvariables -vlan <positive_integer> [-ceaseRouterAdv] [-sendRouterAdv] [-srcLinkLayerAddrOption] [-onlyUnicastRtAdvResponse] [-managedAddrConfig] [-otherAddrConfig] [-currHopLimit] [-maxRtAdvInterval] [-minRtAdvInterval] [-linkMTU] [-reachableTime] [-retransTime] [-defaultLifeTime]
```

show nd6RAvariables

Display Router Advertisement configuration variables.

Synopsys

```
show nd6RAvariables [-vlan <positive_integer>]
```

Arguments

vlan

The VLAN number.

Minimum value: 0

Maximum value: 4094

Outputs

ceaseRouterAdv

Cease router advertisements on this vlan.

sendRouterAdv

whether the router sends periodic RAs and responds to Router Solicitations.

srcLinkLayerAddrOption

Include source link layer address option in RA messages.

onlyUnicastRtAdvResponse

Send only Unicast Router Advertisements in respond to Router Solicitations.

managedAddrConfig

Value to be placed in the Managed address configuration flag field.

otherAddrConfig

Value to be placed in the Other configuration flag field.

currHopLimit

Current Hop limit.

maxRtAdvInterval

Maximum time allowed between unsolicited multicast RAs, in seconds.

minRtAdvInterval

Minimum time interval between RA messages, in seconds.

linkMTU

The Link MTU.

reachableTime

Reachable time, in milliseconds.

retransTime

Retransmission time, in milliseconds.

defaultLifeTime

Default life time, in seconds.

stateflag

RA Param state flags.

lastRtAdvTime

Last RA sent timestamp.

nextRtAdvDelay

Next RA delay.

ipv6Prefix

Onlink prefixes for RA messages.

devno**count**

bind nd6RAvariables

Bind on-link global prefixes to Router Advertisements variables.

Synopsys

```
bind nd6RAvariables -vlan <positive_integer> -ipv6Prefix <ipv6_addr|*>
```

Arguments

vlan

The VLAN number.

Minimum value: 0

Maximum value: 4094

ipv6Prefix

Onlink prefixes for RA messages.

Example

```
bind nd6RAvariables -vlan 2 -ipv6Prefix 8000::/64
```

unbind nd6RAvariables

Unbind prefix from Router Advertisement parameters in NetScaler

Synopsys

```
unbind nd6RAvariables -vlan <positive_integer> -ipv6Prefix <ipv6_addr|*>
```

Arguments

vlan

The VLAN number.

Minimum value: 0

Maximum value: 4094

ipv6Prefix

Onlink prefixes for RA messages.

Example

```
unbind nd6RAvariables -vlan 2 -ipv6Prefix 8000::/64
```

netProfile

The following operations can be performed on "netProfile":

add | **rm** | **set** | **unset** | **show**

add netProfile

Creates a net profile. A net profile (or network profile) contains an IP address or an IP set. During communication with physical servers or peers, the NetScaler appliance uses the addresses specified in the profile as the source IP address.

Synopsys

```
add netProfile <name> [-td <positive_integer>] [-srcIP <string>] [-srcippersistency ( ENABLED | DISABLED )]
```

Arguments

name

Name for the net profile. Must begin with a letter, number, or the underscore character (_), and can consist of letters, numbers, and the hyphen (-), period (.), pound (#), space (), at sign (@), equals (=), colon (:), and underscore characters. Cannot be changed after the profile is created. Choose a name that helps identify the net profile.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

srcIP

IP address or the name of an IP set.

srcippersistency

When the net profile is associated with a virtual server or its bound services, this option enables the NetScaler appliance to use the same address, specified in the net profile, to communicate to servers for all sessions initiated from a particular client to the virtual server.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
add netProfile prof1 -srcip 10.102.1.10
```

rm netProfile

Removes a net profile from the NetScaler appliance.

Synopsys

```
rm netProfile <name> ...
```

Arguments

name

Name of the net profile to be removed.

Example

```
rm netProfile prof1
```

set netProfile

Modifies the srcIP parameter of a net profile.

Synopsys

```
set netProfile <name> [-srcIP <string>] [-srcippersistency ( ENABLED | DISABLED )]
```

Arguments

name

Name of the net profile whose parameter you want to modify.

srcIP

IP address or the name of an IP set.

srcippersistency

When the net profile is associated with a virtual server or its bound services, this option enables the NetScaler appliance to use the same address, specified in the net profile, to communicate to servers for all sessions initiated from a particular client to the virtual server.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
set netProfile prof_1 -srcIP 10.102.1.10
```

unset netProfile

Removes the srcIP attribute of a net profile..Refer to the set netProfile command for meanings of the arguments.

Synopsys

```
unset netProfile <name> [-srcIP] [-srcippersistency]
```

Example

```
unset netProfile prof1 -srcIP
```

show netProfile

Displays the settings of all net profiles configured on the NetScaler appliance, or of the specified net profile.

Synopsys

```
show netProfile [<name>]
```

Arguments

name

Name of the net profile whose details you want to display.

Outputs

srcIP

Source IPaddress or IPSET name.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

srcippersistency

When the net profile is associated with a virtual server or its bound services, this option enables the NetScaler appliance to use the same address, specified in the net profile, to communicate to servers for all sessions initiated from a particular client to the virtual server.

netprofRefCount

Used to keep reference count of IP

vpathEncap

enable/disable vPath Encapsulation

devno

count

stateflag

Example

```
show netProfile
```

netbridge

The following operations can be performed on "netbridge":

`add` | `rm` | `show` | `bind` | `unbind`

add netbridge

Add a network bridge.

Synopsis

```
add netbridge <name>
```

Arguments

name

The name of the network bridge.

Example

```
add netbridge bridge1
```

rm netbridge

Remove a network bridge.

Synopsis

```
rm netbridge <name>
```

Arguments

name

The name of the network bridge.

Example

```
remove netbridge bridge1
```

show netbridge

Show configured network bridges.

Synopsis

```
show netbridge [<name>]
```

Arguments

name

The name of the network bridge.

Outputs

tunnel

The name of the tunnel that is a part of this bridge.

vlan

The VLAN that is extended by this network bridge.

IPAddress

The subnet that is extended by this network bridge.

netmask

The network mask for the subnet.

stateflag

Used internally for display.

devno**count**

bind netbridge

Bind a network bridge to its attributes.

Synopsys

```
bind netbridge <name> [-tunnel <string> ...] [-vlan <positive_integer> ...] [-IPAddress <ip_addr|ipv6_addr|*> [<netmask>]]
```

Arguments

name

The name of the network bridge.

tunnel

The name of the tunnel that needs to be a part of this network bridge.

vlan

The VLAN that needs to be extended.

Minimum value: 1

Maximum value: 4094

IPAddress

The subnet that needs to be extended.

netmask

Subnet mask in dotted-decimal format. For example: 255.255.255.0. This parameter is required for IPv4.

Example

```
bind netbridge bridge1 -tunnel tun0
```

unbind netbridge

Unbind a network bridge from its attributes.

Synopsys

```
unbind netbridge <name> [-tunnel <string> ...] [-vlan <positive_integer> ...] [-IPAddress <ip_addr|ipv6_addr|*>
[<netmask>]]
```

Arguments

name

The name of the network bridge.

tunnel

The name of the tunnel that is part of this network bridge.

vlan

The vlan that is part of this network bridge.

Minimum value: 1

Maximum value: 4094

IPAddress

The subnet that is part of this network bridge.

netmask

Subnet mask in dotted-decimal format. For example: 255.255.255.0. This parameter is required for IPv4.

Example

```
unbind netbridge bridge1 -tunnel tun0
```

onLinkIPv6Prefix

The following operations can be performed on "onLinkIPv6Prefix":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add onLinkIPv6Prefix

add a new on-link global prefix.

Synopsys

```
add onLinkIPv6Prefix <ipv6Prefix> [-onlinkPrefix ( YES | NO )] [-autonomusPrefix ( YES | NO )] [-depricatePrefix ( YES | NO )] [-decrementPrefixLifeTimes ( YES | NO )] [-prefixValideLifeTime <positive_integer>] [-prefixPreferredLifeTime <positive_integer>]
```

Arguments

ipv6Prefix

Onlink prefixes for RA messages.

onlinkPrefix

RA Prefix onlink flag.

Possible values: YES, NO

Default value: YES

autonomusPrefix

RA Prefix Autonomus flag.

Possible values: YES, NO

Default value: YES

depricatePrefix

Depricate the prefix.

Possible values: YES, NO

Default value: NO

decrementPrefixLifeTimes

RA Prefix Autonomus flag.

Possible values: YES, NO

Default value: NO

prefixValideLifeTime

Valide life time of the prefix, in seconds.

Default value: 2592000

Minimum value: 0

prefixPreferredLifeTime

Preferred life time of the prefix, in seconds.

Default value: 604800

Minimum value: 0

Example

```
add onLinkIPv6Prefix 8000::/64
```

rm onLinkIPv6Prefix

remove an existing on-link global prefix.

Synopsis

```
rm onLinkIPv6Prefix <ipv6Prefix>
```

Arguments

ipv6Prefix

Onlink prefixes for RA messages.

Example

```
rm onLinkIPv6Prefix 8000::/64
```

set onLinkIPv6Prefix

set on-link global prefix's configuration variables.

Synopsis

```
set onLinkIPv6Prefix <ipv6Prefix> [-onlinkPrefix ( YES | NO )] [-autonomusPrefix ( YES | NO )] [-depricatePrefix ( YES | NO )] [-decrementPrefixLifeTimes ( YES | NO )] [-prefixValideLifeTime <positive_integer>] [-prefixPreferredLifeTime <positive_integer>]
```

Arguments

ipv6Prefix

Onlink prefixes for RA messages.

onlinkPrefix

RA Prefix onlink flag.

Possible values: YES, NO

Default value: YES

autonomusPrefix

RA Prefix Autonomus flag.

Possible values: YES, NO

Default value: YES

depricatePrefix

Depricate the prefix.

Possible values: YES, NO

Default value: NO

decrementPrefixLifeTimes

RA Prefix Autonomus flag.

Possible values: YES, NO

Default value: NO

prefixValideLifeTime

Valide life time of the prefix, in seconds.

Default value: 2592000

Minimum value: 0

prefixPreferredLifeTime

Preferred life time of the prefix, in seconds.

Default value: 604800

Minimum value: 0

Example

```
set onLinkIPv6Prefix 8000::/64 -prefixValideLifeTime 2592000
```

unset onLinkIPv6Prefix

Use this command to remove onLinkIPv6Prefix settings. Refer to the set onLinkIPv6Prefix command for meanings of the arguments.

Synopsys

```
unset onLinkIPv6Prefix <ipv6Prefix> [-onlinkPrefix] [-autonomusPrefix] [-depricatePrefix] [-decrementPrefixLifeTimes] [-prefixValideLifeTime] [-prefixPreferredLifeTime]
```

show onLinkIPv6Prefix

displays on-link global prefixes.

Synopsys

```
show onLinkIPv6Prefix [<ipv6Prefix>]
```

Arguments

ipv6Prefix

Onlink prefixes for RA messages.

Outputs

onlinkPrefix

RA Prefix onlink flag.

autonomusPrefix

RA Prefix Autonomus flag.

depricatePrefix

Depricate the prefix.

decrementPrefixLifeTimes

RA Prefix Autonomus flag.

prefixValideLifeTime

Valide life time of the prefix, in seconds.

prefixPreferredLifeTime

Preferred life time of the prefix, in seconds.

stateflag

RA Param state flags

prefixCurrValideLfT

Prefix current valid life time

prefixCurrPreferredLfT

Prefix current preferred life time

devno

count

ptp

The following operations can be performed on "ptp":

[set](#) | [show](#)

set ptp

Specifies whether to use Precision Time Protocol (PTP) to synchronize time across cluster nodes. This command is applicable in a cluster setup only. If you do not want to use PTP, you must disable PTP, by using this command, and instead enable NTP.

Synopsys

```
set ptp -state ( DISABLE | ENABLE )
```

Arguments

state

Enables or disables Precision Time Protocol (PTP) on the appliance. If you disable PTP, make sure you enable Network Time Protocol (NTP) on the cluster.

Possible values: DISABLE, ENABLE

Default value: ENABLE

show ptp

Displays the status of Precision Time Protocol (PTP) on the appliance.

Synopsys

```
show ptp
```

Outputs

state

Enables or disables Precision Time Protocol (PTP) on the appliance. If you disable PTP, make sure you enable Network Time Protocol (NTP) on the cluster.

rnat

The following operations can be performed on "rnat":

clear | **set** | **unset** | **stat** | **show**

clear rnat

Removes an RNAT rule from the NetScaler appliance.

Synopsys

```
clear rnat (((<network> [<netmask>]) | (<aclname> [-redirectPort])) [-natIP <ip_addr|*>@ ...] [-td <positive_integer>]
```

Arguments

network

The network address defined for the RNAT entry.

netmask

The subnet mask for the network address.

aclname

An extended ACL defined for the RNAT entry.

redirectPort

The port number to which the packets are redirected.

natIP

The NAT IP address defined for the RNAT entry.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

set rnat

Modifies parameters of an RNAT rule.

Synopsys

```
set rnat (((<network> [<netmask>] [-natIP <ip_addr|*>@ ...]) | (<aclname> [-redirectPort <port>] [-natIP <ip_addr|*>@ ...])) [-td <positive_integer>] [-srcippersistency ( ENABLED | DISABLED )]
```

Arguments

network

IPv4 network address on whose traffic you want the NetScaler appliance to do RNAT processing.

netmask

Subnet mask associated with the network address.

natIP

The NAT IP(s) assigned to the RNAT.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

aclname

Name of any configured extended ACL whose action is ALLOW. The condition specified in the extended ACL rule is used as the condition for the RNAT6 rule.

redirectPort

The port number to which the packets are redirected.

Minimum value: 1

Maximum value: 65535

srcippersistency

Enables the NetScaler appliance to use the same NAT IP address for all RNAT sessions initiated from a particular server.

Possible values: ENABLED, DISABLED

Default value: DISABLED

unset rnat

Use this command to modify the parameters of configured Reverse NAT on the system..Refer to the set rnat command for meanings of the arguments.

Synopsis

```
unset rnat (((<network> [<netmask>]) | (<aclname> [-redirectPort])) [-td <positive_integer>] [-natIP <ip_addr|*>@ ...] [-srcippersistency]
```

stat rnat

Display statistics for rnat sessions.

Synopsis

```
stat rnat [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

Bytes Received (rnatRxBytes)

Bytes received during RNAT sessions.

Bytes Sent (rnatTxBytes)

Bytes sent during RNAT sessions.

Packets Received (rnatRxPkts)

Packets received during RNAT sessions.

Packets Sent (rnatTxPkts)

Packets sent during RNAT sessions.

Syn Sent (rnatTxSyn)

Requests for connections sent during RNAT sessions.

Current RNAT sessions (rnatSessions)

Currently active RNAT sessions.

Example

```
stat rnat
```

show rnat

Display the Reverse NAT configuration.

Synopsys

```
show rnat
```

Outputs

network

The network address.

netmask

Subnet mask associated with the network address.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

natIP

Nat IP Address.

aclname

Name of any configured extended ACL whose action is ALLOW. The condition specified in the extended ACL rule is used as the condition for the RNAT6 rule.

redirectPort

The port number to which the packets are redirected.

srcippersistency

Enables the NetScaler appliance to use the same NAT IP address for all RNAT sessions initiated from a particular server.

cfgflags

This contains the flags for RNAT in DB

devno

count

stateflag

rnat6

The following operations can be performed on "rnat6":

add | **bind** | **unbind** | **set** | **unset** | **clear** | **show**

add rnat6

Adds a Reverse Network Address Translation (RNAT6) rule for IPv6 traffic. When an IPv6 packet generated by a server matches the conditions specified in the RNAT6 rule, the appliance replaces the source IPv6 address of the IPv6 packet with a configured NAT IPv6 address before forwarding it to the destination.

Synopsys

```
add rnat6 <name> (<network> | (<acl6name> [-redirectPort <port>])) [-td <positive_integer>] [-srcippersistency (
ENABLED | DISABLED )]
```

Arguments

name

Name for the RNAT6 rule. Must begin with a letter, number, or the underscore character (_), and can consist of letters, numbers, and the hyphen (-), period (.), pound (#), space (), at sign (@), equals (=), colon (:), and underscore characters. Cannot be changed after the rule is created. Choose a name that helps identify the RNAT6 rule.

network

IPv6 address of the network on whose traffic you want the NetScaler appliance to do RNAT processing.

acl6name

Name of any configured ACL6 whose action is ALLOW. The rule of the ACL6 is used as an RNAT6 rule.

redirectPort

Port number to which the IPv6 packets are redirected. Applicable to TCP and UDP protocols.

Minimum value: 1

Maximum value: 65535

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

srcippersistency

Enable source ip persistency, which enables the NetScaler appliance to use the RNAT ips using source ip.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
add rnat6 rnat6_name 2002::/64
```

bind rnat6

Binds specified IPv6 NAT IPs to an RNAT6 rule.

Synopsys

```
bind rnat6 <name> <natIP6>@ ...
```

Arguments

name

Name of the RNAT6 rule to which to bind NAT IPs.

natIP6

One or more IP addresses to be bound to the IP set.

Example

```
bind rnat6 <rnat6_name> <natIP6>@ ...
```

unbind rnat6

Unbinds the associated NAT IPv6 address(es) from an RNAT6 rule.

Synopsys

```
unbind rnat6 <name> <natIP6>@ ...
```

Arguments

name

Name of the RNAT6 rule from which to unbind the associated NAT IP address(es).

natIP6

IP address, or multiple addresses, to be unbound from the RNAT6rule. (If using the CLI, use spaces to separate multiple addresses.)

Example

```
unbind rnat6 <rnat6_name> <natIP6>@ ...
```

set rnat6

Modifies the specified parameters of an RNAT6 rule.

Synopsys

```
set rnat6 <name> [-redirectPort <port>] [-srcippersistency ( ENABLED | DISABLED )]
```

Arguments

name

Name of the RNAT6 rule. Required for identifying the RNAT6 rule and cannot be modified.

redirectPort

Port number to which the IPv6 packets are redirected. Applicable to TCP and UDP protocols.

Minimum value: 1

Maximum value: 65535

srcippersistency

Enable source ip persistency, which enables the NetScaler appliance to use the RNAT6 ips using source ip.

Possible values: ENABLED, DISABLED

Default value: DISABLED

unset rnat6

Resets the specified parameters of an RNAT6 rule to their default settings. Refer to the set rnat6 command for parameter descriptions..Refer to the set rnat6 command for meanings of the arguments.

Synopsys

```
unset rnat6 <name> [-redirectPort] [-srcippersistency]
```

clear rnat6

Removes an RNAT6 rule from the NetScaler appliance.

Synopsys

```
clear rnat6 <name>
```

Arguments

name

Name of the RNAT6 rule to be removed.

show rnat6

Displays the settings of all RNAT6 rules configured on the NetScaler appliance, or of the specified RNAT6 rule.

Synopsys

```
show rnat6 [<name>]
```

Arguments

name

Name of the RNAT6 rule whose details you want to display.

Outputs

network

The network address.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

acl6name

ACL6 name

natIP6

Nat IP Address.

redirectPort

Redirect Port Value

srcippersistency

Enable source ip persistency, which enables the NetScaler appliance to use the RNAT6 ips using source ip.

stateflag

devno

count

rnatglobal

The following operations can be performed on "rnatglobal":

[show](#) | [bind](#) | [unbind](#)

show rnatglobal

Display the Reverse NAT configuration.

Synopsys

show rnatglobal

Outputs

policy

The policy Name.

stateflag

priority

The priority of the policy.

devno

count

bind rnatglobal

Bind rnat to policy for logging purpose

Synopsys

bind rnatglobal [-policy <string> [-priority <positive_integer>]]

Arguments

policy

Name of the policy getting bound to the RNAT globally. This policy will apply to all the RNATS present

priority

Priority of the policy

Minimum value: 0

unbind rnatglobal

Unbind policy from rnat

Synopsys

unbind rnatglobal (-policy <string> | -all)

Arguments

policy

Name of the policy to be unbound from the RNAT globally.

all

Remove all RNAT global config

rnatip

The following operations can be performed on "rnatip":

stat rnatip

Display statistics for RNAT sessions.

Synopsys

```
stat rnatip [<rnatip>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

rnatip

Specifies the NAT IP address of the configured RNAT entry for which you want to see the statistics. If you do not specify an IP address, this displays the statistics for all the configured RNAT entries.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Traffic domain (td)

Traffic domain for ipaddr.

Bytes Received (rxBytes)

Bytes received on this IP address during RNAT sessions.

Bytes Sent (txBytes)

Bytes sent from this IP address during RNAT sessions.

Packets Received (rxPkts)

Packets received on this IP address during RNAT sessions.

Packets Sent (txPkts)

Packets sent from this IP address during RNAT sessions.

Syn Sent (txSyn)

Requests for connections sent from this IP address during RNAT sessions.

Current RNAT sessions (sessions)

Currently active RNAT sessions started from this IP address.

Example

```
stat rnatip 1.1.1.1
```

rnatparam

The following operations can be performed on "rnatparam":

[set](#) | [unset](#) | [show](#)

set rnatparam

Sets global parameters of RNAT rules on the NetScaler appliance.

Synopsys

```
set rnatparam [-tcpproxy ( ENABLED | DISABLED )] [-srcippersistency ( ENABLED | DISABLED )]
```

Arguments

tcpproxy

Enable TCP proxy, which enables the NetScaler appliance to optimize the RNAT TCP traffic by using Layer 4 features.

Possible values: ENABLED, DISABLED

Default value: ENABLED

srcippersistency

Enable source ip persistency, which enables the NetScaler appliance to use the RNAT ips using source ip.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
set rnatparam -tcpproxy ENABLED or set rnatparam -srcippersistency ENABLED.
```

unset rnatparam

Use this command to remove rnatparam settings.Refer to the set rnatparam command for meanings of the arguments.

Synopsys

```
unset rnatparam [-tcpproxy] [-srcippersistency]
```

show rnatparam

Show the rnat parameter.

Synopsys

```
show rnatparam
```

Outputs

tcpproxy

Enable TCP proxy, which enables the NetScaler appliance to optimize the RNAT TCP traffic by using Layer 4 features.

srcippersistency

Enable source ip persistency, which enables the NetScaler appliance to use the RNAT ips using source ip.

Example

```
show rnat parameter
```


route

The following operations can be performed on "route":

add | **clear** | **rm** | **set** | **unset** | **show**

add route

Adds an IPv4 static route to the routing table of the NetScaler appliance.

Synopsys

```
add route <network> <netmask> <gateway> [-td <positive_integer>] [-distance <positive_integer>] [-cost  
<positive_integer>] [-weight <positive_integer>] [-advertise ( DISABLED | ENABLED )] [-protocol <protocol> ...] [-msr  
( ENABLED | DISABLED )] [-monitor <string>]]
```

Arguments

network

IPv4 network address for which to add a route entry in the routing table of the NetScaler appliance.

netmask

The subnet mask associated with the network address.

gateway

IP address of the gateway for this route. Can be either the IP address of the gateway, or can be null to specify a null interface route.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

distance

Administrative distance of this route, which determines the preference of this route over other routes, with same destination, from different routing protocols. A lower value is preferred.

Default value: 1

Minimum value: 0

Maximum value: 255

cost

The cost of a route is used to compare routes of the same type. The route having the lowest cost is the most preferred route. Possible values: 0 through 65535. Default: 0.

Minimum value: 0

Maximum value: 65535

weight

Positive integer used by the routing algorithms to determine preference for this route over others of equal cost. The lower the weight, the higher the preference.

Default value: 1

Minimum value: 1

Maximum value: 65535

advertise

Advertise this route.

Possible values: DISABLED, ENABLED

protocol

Routing protocol used for advertising this route.

Default value: ADV_ROUTE_FLAGS

msr

Monitor this route using a monitor of type ARP or PING.

Possible values: ENABLED, DISABLED

Default value: DISABLED

monitor

Name of the monitor, of type ARP or PING, configured on the NetScaler appliance to monitor this route.

Example

```
add route 10.10.10.0 255.255.255.0 10.10.10.1
```

clear route

Removes routes of the specified type(protocol) from the routing table of the NetScaler appliance.

Synopsis

```
clear route <routeType>
```

Arguments

routeType

Protocol used by routes that you want to remove from the routing table of the NetScaler appliance.

rm route

Removes a static route from the NetScaler appliance. Note: You cannot use this command to remove routes that are part of a VLAN configuration. Use the `rmvlan` or `clear vlan` command instead.

Synopsis

```
rm route <network> <netmask> <gateway> [-td <positive_integer>]
```

Arguments

network

Network address specified in the route entry that you want to remove from the routing table of the NetScaler appliance.

netmask

Subnet mask associated with the network address.

gateway

IP address of the gateway for this route.

td

The Traffic Domain Id of the route to be removed.

Minimum value: 0

Maximum value: 4094

set route

Modifies parameters of an IPv4 static route.

Synopsys

```
set route <network> <netmask> <gateway> [-td <positive_integer>] [-distance <positive_integer>] [-cost  
<positive_integer>] [-weight <positive_integer>] [-advertise ( DISABLED | ENABLED )] [-protocol <protocol> ...] [-msr  
( ENABLED | DISABLED )] [-monitor <string>]]
```

Arguments

network

Network address in the route entry that you want to modify.

netmask

Subnet mask associated with the network address.

gateway

IP address of the gateway for this route. Can be either the IP address of the gateway, or can be null to specify a null interface route.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

distance

Administrative distance of this route, which determines the preference of this route over other routes, with same destination, from different routing protocols. A lower value is preferred.

Default value: 1

Minimum value: 0

Maximum value: 255

cost

The cost of a route is used to compare routes of the same type. The route having the lowest cost is the most preferred route. Possible values: 0 through 65535. Default: 0.

Minimum value: 0

Maximum value: 65535

weight

Positive integer used by the routing algorithms to determine preference for this route over others of equal cost. The lower the weight, the higher the preference.

Default value: 1

Minimum value: 1

Maximum value: 65535

advertise

Advertise this route.

Possible values: DISABLED, ENABLED

protocol

Routing protocol used for advertising this route.

Default value: ADV_ROUTE_FLAGS

msr

Monitor this route using a monitor of type ARP or PING.

Possible values: ENABLED, DISABLED

Default value: DISABLED

monitor

Name of the monitor, of type ARP or PING, configured on the NetScaler appliance to monitor this route.

Example

```
set route 10.10.10.0 255.255.255.0 10.10.10.1 -advertise enable
```

unset route

Unset the attributes of a route that were added by the add/set route command..Refer to the set route command for meanings of the arguments.

Synopsys

```
unset route <network> <netmask> <gateway> [-td <positive_integer>] [-advertise] [-distance] [-cost] [-weight] [-protocol] [-msr] [-monitor]
```

Example

```
unset route 10.10.10.0 255.255.255.0 10.10.10.1 -advertise enable
```

show route

Display the configured routing information.

Synopsys

```
show route [<network> <netmask> [<gateway>] [-td <positive_integer>]] [<routeType>] [-detail]
```

Arguments

network

The destination network or host.

netmask

The subnet mask associated with the network address.

gateway

The gateway for the route.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

routeType

The type of routes to be shown.

detail

Display a detailed view.

Outputs

gatewayName

The name of the gateway for this route. For a route other than a link load balancing (LLB) route, this value is null.

advertise

Enable advertisement.

type

State of the RNAT.

stateflag**dynamic**

State of the route.

STATIC**PERMANENT****DIRECT****NAT****LBROUTE****ADV****TUNNEL**

Show whether it is a tunnel route or not.

cost

The cost of a route is used to compare routes of the same type. The route having the lowest cost is the most preferred route. Possible values: 0 through 65535. Default: 0.

distance

Administrative distance of this route, which determines the preference of this route over other routes, with same destination, from different routing protocols. A lower value is preferred.

weight

The weight of this route.

protocol

Routing protocol used for advertising this route.

data

Internal data of this route.

data0

Internal route type is stored, used for get.

flags

If this route is dynamic, the name of the routing protocol from which it was learned.

routeOwners

Use this option with -dynamic and in a cluster only to specify the set of nodes from which this dynamic route has been learnt.

retain

OSPF

OSPF protocol.

ISIS

ISIS protocol.

RIP

RIP protocol.

BGP

BGP protocol.

DHCP

advOSPF

Advertised through OSPF protocol.

advISIS

Advertised through ISIS protocol.

advRIP

Advertised through RIP protocol.

advBGP

Advertised through BGP protocol.

msr

Whether MSR is enabled or disabled.

monitor

Name of the monitor, of type ARP or PING, configured on the NetScaler appliance to monitor this route.

state

The state of the static route. Possible values: UP, DOWN.

peFlags

PE flags.

totalprobes

The total number of probes sent.

totalfailedprobes

The total number of failed probes.

failedprobes

Number of the current failed monitoring probes.

monStatCode

The code indicating the monitor response.

monStatParam1

First parameter used with the message code.

monStatParam2

Second parameter used with the message code.

monStatParam3

Third parameter used with the message code.

devno**count**

Example

An example of the output of the show route command is as follows: 3 configured routes:

route6

The following operations can be performed on "route6":

add | **clear** | **rm** | **set** | **unset** | **show**

add route6

Adds an IPv6 static route to the routing table of the NetScaler appliance.

Synopsys

```
add route6 <network> [<gateway>] [-vlan <positive_integer>] [-weight <positive_integer>] [-distance  
<positive_integer>] [-cost <positive_integer>] [-advertise ( DISABLED | ENABLED )] [-msr ( ENABLED | DISABLED  
)] [-monitor <string>]] [-td <positive_integer>]
```

Arguments

network

IPv6 network address for which to add a route entry to the routing table of the NetScaler appliance.

gateway

The gateway for this route. The value for this parameter is either an IPv6 address or null.

Default value: 0

vlan

Integer value that uniquely identifies a VLAN through which the NetScaler appliance forwards the packets for this route.

Default value: 0

Minimum value: 0

Maximum value: 4094

weight

Positive integer used by the routing algorithms to determine preference for this route over others of equal cost. The lower the weight, the higher the preference.

Default value: 1

Minimum value: 1

Maximum value: 65535

distance

Administrative distance of this route from the appliance.

Default value: 1

Minimum value: 1

Maximum value: 254

cost

Positive integer used by the routing algorithms to determine preference for this route. The lower the cost, the higher the preference.

Default value: 1

Minimum value: 0

Maximum value: 65535

advertise

Advertise this route.

Possible values: DISABLED, ENABLED

msr

Monitor this route with a monitor of type ND6 or PING.

Possible values: ENABLED, DISABLED

Default value: DISABLED

monitor

Name of the monitor, of type ND6 or PING, configured on the NetScaler appliance to monitor this route.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

Example

```
add route6 ::/0 2004::1 add route6 ::/0 FE80::67 -vlan 5
```

clear route6

Removes IPv6 routes of the specified type (protocol) from the routing table of the NetScaler appliance.

Synopsis

```
clear route6 <routeType>
```

Arguments

routeType

Type of IPv6 routes to remove from the routing table of the NetScaler appliance.

rm route6

Removes a static IPv6 route from the NetScaler appliance.

Synopsis

```
rm route6 <network> [<gateway>] [-vlan <positive_integer>] [-td <positive_integer>]
```

Arguments

network

The network of the route to be removed.

gateway

The gateway address of the route to be removed.

Default value: 0

vlan

Integer that uniquely identifies the VLAN defined for this route.

Default value: 0

Minimum value: 0

Maximum value: 4094

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

Example

```
rm route6 ::/0 2004::1 rm route6 ::/0 FE80::67 -vlan 5
```

set route6

Modifies parameters of an IPv6 static route.

Synopsys

```
set route6 <network> [<gateway>] [-vlan <positive_integer>] [-weight <positive_integer>] [-distance  
<positive_integer>] [-cost <positive_integer>] [-advertise ( DISABLED | ENABLED )] [-msr ( ENABLED | DISABLED  
)] [-monitor <string>]] [-td <positive_integer>]
```

Arguments

network

IPv6 network address of the route entry to be modified.

gateway

The gateway for the route's destination network.

Default value: 0

vlan

Integer value that uniquely identifies a VLAN through which the NetScaler appliance forwards the packets for this route.

Default value: 0

Minimum value: 0

Maximum value: 4094

weight

Positive integer used by the routing algorithms to determine preference for this route over others of equal cost. The lower the weight, the higher the preference.

Default value: 1

Minimum value: 1

Maximum value: 65535

distance

Administrative distance of this route from the appliance.

Default value: 1

Minimum value: 1

Maximum value: 254

cost

Positive integer used by the routing algorithms to determine preference for this route. The lower the cost, the higher the preference.

Default value: 1

Minimum value: 0

Maximum value: 65535

advertise

Advertise this route.

Possible values: DISABLED, ENABLED

msr

Monitor this route with a monitor of type ND6 or PING.

Possible values: ENABLED, DISABLED

Default value: DISABLED

monitor

Name of the monitor, of type ND6 or PING, configured on the NetScaler appliance to monitor this route.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

Example

```
set route6 1::1/100 2000::1 -advertise enable
```

unset route6

Unset the attributes of a route that were added by the add/set route command. Refer to the set route6 command for meanings of the arguments.

Synopsis

```
unset route6 <network> [<gateway>] [-vlan <positive_integer>] [-td <positive_integer>] [-weight] [-distance] [-cost] [-advertise] [-msr] [-monitor]
```

Example

```
unset route6 2000::1/100 3000::1 -advertise enable
```

show route6

Displays configuration and state information of all IPv6 routes in the NetScaler appliance's routing table, or of the specified IPv6 route.

Synopsys

```
show route6 [<network> [<gateway>] [-vlan <positive_integer>] [-td <positive_integer>]] [<routeType>] [-detail]
```

Arguments

network

IPv6 network address of the route entry for which to display details.

gateway

Any gateway of the route entry for which to display details.

Default value: 0

vlan

Integer that uniquely identifies the VLAN defined for this route.

Default value: 0

Minimum value: 0

Maximum value: 4094

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

routeType

The type of IPv6 routes to be to be displayed.

detail

To get a detailed view.

Outputs

gatewayName

The name of the gateway for this route.

advertise

Any gateway of the route entry for which the details are to be displayed.

type

State of the RNAT.

stateflag

dynamic

Whether this route is dynamically learned or not.

weight

Weight of this route.

distance

Administrative distance of this route from the appliance.

cost

Positive integer used by the routing algorithms to determine preference for this route. The lower the cost, the higher the preference.

data

Internal data of this route.

flags

For a dynamic route, the routing protocol from which the route was learned.

msr

Whether MSR is enabled or disabled.

monitor

Name of the monitor, of type ND6 or PING, configured on the NetScaler appliance to monitor this route.

state

Whether this route is UP or DOWN.

totalprobes

The total number of probes sent.

totalfailedprobes

The total number of failed probes.

failedprobes

Current number of failed monitoring probes.

monStatCode

The code indicating the monitor response.

monStatParam1

First parameter for use with message code.

monStatParam2

Second parameter for use with message code.

monStatParam3

Third parameter for use with message code.

data1

Internal data of this route.

routeOwners

Use this option with -dynamic and in a cluster only to specify the set of nodes from which this dynamic route has been learnt.

retain**STATIC**

Static route.

PERMANENT

Permanent Route.

connected

Connected Route.

OSPFV3

For a dynamic route, the routing protocol from which the route was learned.

ISIS

If this route is dynamic then which routing protocol was it learnt from.

active

For a dynamic route, the routing protocol from which the route was learned.

BGP

For a dynamic route, the routing protocol from which the route was learned.

RIP

For a dynamic route, the routing protocol from which the route was learned.

raRoute

For a dynamic route, the routing protocol from which the route was learned.

devno

count

Example

Following is an example of the output of the show route6 command:

Flags: Static(S),

rsskeytype

The following operations can be performed on "rsskeytype":

[set](#) | [show](#)

set rsskeytype

Synopsys

```
set rsskeytype -rsstype ( ASYMMETRIC | SYMMETRIC )
```

Arguments

rsstype

Type of RSS key, possible values ASYMMETRIC and SYMMETRIC.

Possible values: ASYMMETRIC, SYMMETRIC

Default value: ASYMMETRIC

show rsskeytype

Synopsys

```
show rsskeytype
```

Outputs

rsstype

Type of RSS key, possible values ASYMMETRIC and SYMMETRIC.

tunnelip

The following operations can be performed on "tunnelip":

stat tunnelip

Display the statistics related to IP tunnel.

Synopsys

stat tunnelip [<tunnelip>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

tunnelip

remote IP address of the configured tunnel.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Packets received on tunnel (tnIRxPkts)

Total number of packets received on the tunnel.

Packets transmitted on tunnel (tnITxPkts)

Total number of packets transmitted on the tunnel.

Bytes received on tunnel (tnIRxBytes)

Total number of bytes received on the tunnel.

Bytes transmitted on tunnel (tnITxBytes)

Total number of bytes transmitted on the tunnel.

Example

```
stat tunnelip 2.1.1.1
```

tunnelip6

The following operations can be performed on "tunnelip6":

stat tunnelip6

Display the statistics related to IP tunnel.

Synopsys

```
stat tunnelip6 [<tunnelip6>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full )]
```

Arguments

tunnelip6

remote IPv6 address of the configured tunnel.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Packets received on tunnel (tnlRxPkts)

Total number of packets received on the tunnel.

Packets transmitted on tunnel (tnlTxPkts)

Total number of packets transmitted on the tunnel.

Bytes received on tunnel (tnlRxBytes)

Total number of bytes received on the tunnel.

Bytes transmitted on tunnel (tnITxBytes)

Total number of bytes transmitted on the tunnel.

Example

```
stat tunnelip6 2001::1
```

vPathParam

The following operations can be performed on "vPathParam":

[set](#) | [unset](#) | [show](#)

set vPathParam

Sets the global parameters for vPath

Synopsis

```
set vPathParam [-srcIP <ip_addr>] [-offload ( ENABLED | DISABLED )]
```

Arguments

srcIP

source-IP address used for all vPath L3 encapsulations. Must be a MIP or SNIP address.

offload

enable/disable vPath offload feature

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
set vpathparam -srcip 2.2.2.2
```

unset vPathParam

Use this command to remove vPathParam settings. Refer to the set vPathParam command for meanings of the arguments.

Synopsis

```
unset vPathParam [-srcIP] [-offload]
```

show vPathParam

Display the global parameters for vPath

Synopsis

```
show vPathParam
```

Outputs

srcIP

srcIP used for vPath encapsulation.

Encapsulation

Global vPath encapsulation .

offload

enable/disable vPath offload feature

Example

```
show vpathparam
```

vlan

The following operations can be performed on "vlan":

add | **rm** | **set** | **unset** | **bind** | **unbind** | **show** | **stat**

add vlan

Adds a VLAN to the NetScaler appliance. The new VLAN is not active unless interfaces are bound to it.

Synopsys

```
add vlan <id> [-aliasName <string>] [-ipv6DynamicRouting ( ENABLED | DISABLED )] [-mtu <positive_integer>]
```

Arguments

id

A positive integer that uniquely identifies a VLAN.

Minimum value: 1

Maximum value: 4094

aliasName

A name for the VLAN. Must begin with a letter, a number, or the underscore symbol, and can consist of from 1 to 31 letters, numbers, and the hyphen (-), period (.), pound (#), space (), at sign (@), equals (=), colon (:), and underscore (_) characters. You should choose a name that helps identify the VLAN. However, you cannot perform any VLAN operation by specifying this name instead of the VLAN ID.

ipv6DynamicRouting

Enable all IPv6 dynamic routing protocols on this VLAN. Note: For the ENABLED setting to work, you must configure IPv6 dynamic routing protocols from the VTYSH command line.

Possible values: ENABLED, DISABLED

Default value: DISABLED

mtu

Specifies the maximum transmission unit (MTU), in bytes. The MTU is the largest packet size, excluding 14 bytes of ethernet header and 4 bytes of crc, that can be transmitted and received over this VLAN.

Default value: 0

Minimum value: 500

Maximum value: 9216

rm vlan

Removes a VLAN from the NetScaler appliance. When the VLAN is removed, its interfaces are bound to VLAN 1. Note: VLAN 1 cannot be removed by any command.

Synopsys

```
rm vlan <id>
```

Arguments

id

Integer that uniquely identifies the VLAN to be removed from the NetScaler appliance. When the VLAN is removed, its interfaces become members of VLAN 1.

Minimum value: 2

Maximum value: 4094

set vlan

Modifies parameters of a VLAN on the NetScaler appliance.

Synopsys

```
set vlan <id> [-aliasName <string>] [-ipv6DynamicRouting ( ENABLED | DISABLED )] [-mtu <positive_integer>]
```

Arguments

id

A positive integer that uniquely identifies a VLAN.

Minimum value: 1

Maximum value: 4094

aliasName

A name for the VLAN. Must begin with a letter, a number, or the underscore symbol, and can consist of from 1 to 31 letters, numbers, and the hyphen (-), period (.) pound (#), space (), at sign (@), equals (=), colon (:), and underscore (_) characters. You should choose a name that helps identify the VLAN. However, you cannot perform any VLAN operation by specifying this name instead of the VLAN ID.

ipv6DynamicRouting

Enable all IPv6 dynamic routing protocols on this bridge group. Note: For the ENABLED setting to work, you must configure IPv6 dynamic routing protocols from the VTYSH command line.

Possible values: ENABLED, DISABLED

Default value: DISABLED

mtu

Specifies the maximum transmission unit (MTU), in bytes. The MTU is the largest packet size, excluding 14 bytes of ethernet header and 4 bytes of crc, that can be transmitted and received over this VLAN.

Default value: 0

Minimum value: 500

Maximum value: 9216

Example

```
set vlan 2 -dynamicRouting ENABLED
```

unset vlan

Use this command to remove vlan settings.Refer to the set vlan command for meanings of the arguments.

Synopsys

```
unset vlan <id> [-aliasName] [-ipv6DynamicRouting] [-mtu]
```

bind vlan

Binds the specified interfaces or IP addresses to a VLAN. An interface can be bound to a VLAN as a tagged or an untagged member. Adding an interface as an untagged member removes it from its current native VLAN and adds it to the new VLAN. If an interface is added as a tagged member to a VLAN, it still remains a member of its native VLAN.

Synopsys

```
bind vlan <id> [-ifnum <interface_name> ... [-tagged]] [-IPAddress <ip_addr|ipv6_addr|*> [<netmask>] [-td <positive_integer>]]
```

Arguments

id

Specifies the virtual LAN ID.

Minimum value: 1

Maximum value: 4094

ifnum

Interface to be bound to the VLAN, specified in slot/port notation (for example, 1/3).

Minimum value: 1

tagged

Make the interface an 802.1q tagged interface. Packets leaving this interface have an additional 4-byte 802.1q tag, which identifies the VLAN. To use 802.1q tagging, you must also configure the switch connected to the interfaces on the appliance.

IPAddress

Network address to be associated with the VLAN. Should exist on the appliance before you associate it with the VLAN. To enable IP forwarding among VLANs, the specified address can be used as the default gateway by the hosts in the network.

netmask

Subnet mask for the network address defined for this VLAN.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

unbind vlan

Unbinds the specified interfaces or IP addresses from a VLAN. If any of the interfaces are untagged members of the VLAN, they are automatically bound to VLAN 1.

Synopsys

```
unbind vlan <id> [-ifnum <interface_name> ... [-tagged]] [-IPAddress <ip_addr|ipv6_addr|*> [<netmask>] [-td <positive_integer>]]
```

Arguments

id

The virtual LAN (VLAN) id.

Minimum value: 1

Maximum value: 4094

ifnum

Interface to unbind from the VLAN, specified in slot/port notation (for example, 1/3).

Minimum value: 1

tagged

The 802.1q tagged interface.

IPAddress

The IP Address associated with the VLAN configuration.

netmask

Subnet mask for the network address defined for this VLAN.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

show vlan

Displays the settings of all VLANs configured on the NetScaler appliance, or of the specified VLAN. To display the settings of all the VLANs, run the command without any parameters. To display the settings of a particular VLAN, specify the ID of the VLAN.

Synopsys

show vlan [<id>] show vlan stats - alias for 'stat vlan'

Arguments

id

Integer that uniquely identifies the VLAN for which the details are to be displayed.

Minimum value: 1

Maximum value: 4094

Outputs

aliasName

A name for the VLAN. Must begin with a letter, a number, or the underscore symbol, and can consist of from 1 to 31 letters, numbers, and the hyphen (-), period (.) pound (#), space (), at sign (@), equals (=), colon (:), and underscore (_) characters. You should choose a name that helps identify the VLAN. However, you cannot perform any VLAN operation by specifying this name instead of the VLAN ID.

IPAddress

The IP address assigned to the VLAN.

netmask

Subnet mask for the network address defined for this VLAN.

linklocalIPv6Addr

The link-local IP address assigned to the VLAN.

rnat

Temporary flag used for internal purpose.

stateflag

state flag

portbitmap

Member interfaces of this vlan.

lsbitmap

Member linksets of this vlan.

tagbitmap

Tagged members of this vlan.

lstagbitmap

Tagged linksets of this vlan.

ifaces

Names of all member interfaces of this vlan.

taglfaces

Names of all tagged member interfaces of this vlan.

ipv6DynamicRouting

Whether dynamic routing is enabled or disabled.

flag

ifnum

The interface to be bound to the VLAN, specified in slot/port notation (for example, 1/3).

tagged

Make the interface an 802.1q tagged interface. Packets sent on this interface on this VLAN have an additional 4-byte 802.1q tag, which identifies the VLAN. To use 802.1q tagging, you must also configure the switch connected to the appliance's interfaces.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

vlantd

Traffic domain associated with vlan.

sdxVlan

SDX vlan.

mtu

Specifies the maximum transmission unit (MTU), in bytes. The MTU is the largest packet size, excluding 14 bytes of ethernet header and 4 bytes of crc, that can be transmitted and received over this VLAN.

vxlan

The VXLAN that extends this vlan.

devno

count

Example

An example of the output of the show vlan command is as follows: 1) VLAN ID: 5

stat vlan

Display statistics for VLAN(s).

Synopsys

stat vlan [<id>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

id

An integer specifying the VLAN identification number (VID). Possible values: 1 through 4094.

Minimum value: 1

Maximum value: 4094

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Packets received (RxPkts)

Packets received on the VLAN.

Bytes received (RxBytes)

Bytes of data received on the VLAN.

Packets sent (TxPkts)

Packets transmitted on the VLAN.

Bytes sent (TxBytes)

Bytes of data transmitted on the VLAN.

Packets dropped (DropPkts)

Inbound packets dropped by the VLAN upon reception.

received (BcastPkt)

Broadcast packets sent and received on the VLAN.

Example

```
stat vlan 1
```

vpath

The following operations can be performed on "vpath":

[add](#) | [rm](#) | [show](#) | [stat](#)

add vpath

Adds vPath destination IP to which packets need to be vPath injected.

Synopsys

```
add vpath <name> (<destIP> [<netmask>] [<gateway>])
```

Arguments

name

Name for the vPath. Must begin with a letter, number, or the underscore character (_), and can consist of letters, numbers, and the hyphen (-), period (.), pound (#), space (), at sign (@), equals (=), colon (:), and underscore characters. Cannot be changed after the profile is created. Choose a name that helps identify the net profile.

destIP

This is the destination ip, where vPath encapsulated packets needs to be sent

netmask

Subnet mask associated with the destination network.

gateway

Next hop gateway to reach the destination address.

Example

```
add vpath vPath1 -destip 10.102.1.10
```

rm vpath

Remove vPath destination IP.

Synopsys

```
rm vpath <name> ...
```

Arguments

name

Name of the vPath to be removed.

Example

```
rm netProfile profil
```

show vpath

List down all vPath destination IPs.

Synopsys

show vpath [<name>]

Arguments

name

Name of the vPath whose details you want to display.

Outputs

destIP

This is the destination ip, where vPath encapsulated packets needs to be sent

netmask

Subnet mask associated with the destination network.

gateway

Next hop gateway to reach the destination address.

devno

count

stateflag

Example

```
show vpath
```

stat vpath

Display vPath statistics.

Synopsys

```
stat vpath [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

L2 data packets received (TotL2DataRx)

Total number of non-fragmented vPath data packets decapsulated in L2 adjacency

L3 data packets received (TotL3DataRx)

Total number of non-fragmented vPath data packets decapsulated in L3 adjacency

L2 control packets received (TotL2CntrlPkts)

Total number of vPath control packets received in L2 adjacency

L3 control packets received (TotL3CntrlPkts)

Total number of vPath control packets received in L3 adjacency

Fragmented packets received (TotFragPkts)

Total number of vPath fragments received

L2 packets transmitted (TotL2EncapPkts)

Total number of L2 vPath encapsulated packets injected to VEM

L3 packets transmitted (TotL3EncapPkts)

Total number of L3 vPath encapsulated packets injected to VEM

Fragmented packets transmitted (TotFragEncapPkts)

Number of fragmented vPath packets transmitted

Offload packets transmitted (TotOffload)

Number of offloaded vPath packets transmitted

vrID

The following operations can be performed on "vrID":

add | **rm** | **set** | **unset** | **bind** | **unbind** | **show**

add vrID

Adds a VMAC address to the NetScaler appliance. A Virtual MAC address (VMAC) is a floating entity, shared by the nodes in an HA configuration.

Synopsys

```
add vrID <id> [-priority <positive_integer>] [-preemption ( ENABLED | DISABLED )] [-sharing ( ENABLED | DISABLED )] [-tracking <tracking>] [-ownerNode <positive_integer>]
```

Arguments

id

Integer that uniquely identifies the VMAC address. The generic VMAC address is in the form of 00:00:5e:00:01:<VRID>. For example, if you add a VRID with a value of 60 and bind it to an interface, the resulting VMAC address is 00:00:5e:00:01:3c, where 3c is the hexadecimal representation of 60.

Minimum value: 1

Maximum value: 255

priority

Base priority (BP), in an active-active mode configuration, which ordinarily determines the master VIP address.

Default value: 255

Minimum value: 1

Maximum value: 255

preemption

In an active-active mode configuration, make a backup VIP address the master if its priority becomes higher than that of a master VIP address bound to this VMAC address.

If you disable pre-emption while a backup VIP address is the master, the backup VIP address remains master until the original master VIP's priority becomes higher than that of the current master.

Possible values: ENABLED, DISABLED

Default value: ENABLED

sharing

In an active-active mode configuration, enable the backup VIP address to process any traffic instead of dropping it.

Possible values: ENABLED, DISABLED

Default value: DISABLED

tracking

The effective priority (EP) value, relative to the base priority (BP) value in an active-active mode configuration. When EP is set to a value other than None, it is EP, not BP, which determines the master VIP address.

Available settings function as follows:

* NONE - No tracking. EP = BP

* ALL - If the status of all virtual servers is UP, EP = BP. Otherwise, EP = 0.

* ONE - If the status of at least one virtual server is UP, EP = BP. Otherwise, EP = 0.

* PROGRESSIVE - If the status of all virtual servers is UP, EP = BP. If the status of all virtual servers is DOWN, EP = 0. Otherwise EP = BP $(1 - K/N)$, where N is the total number of virtual servers associated with the VIP address and K is the number of virtual servers for which the status is DOWN.

Default: NONE.

Possible values: NONE, ONE, ALL, PROGRESSIVE

Default value: NONE

ownerNode

Assign a cluster node as the owner of this VMAC address. If no owner is configured, owner node is displayed as ALL and one node is dynamically elected as the owner.

Default value: -1

Minimum value: 0

Maximum value: 31

Example

```
add vrID 1
```

rm vrID

Removes a specified VMAC entry or all VMAC entries from the NetScaler appliance.

Synopsys

```
rm vrID (<id> | -all)
```

Arguments

id

Integer value that uniquely identifies the VMAC address.

Minimum value: 1

Maximum value: 255

all

Remove all the configured VMAC addresses from the NetScaler appliance.

set vrID

Modifies parameters related to a VMAC address on the NetScaler appliance.

Synopsys

```
set vrID <id> [-priority <positive_integer>] [-preemption ( ENABLED | DISABLED )] [-sharing ( ENABLED | DISABLED )] [-tracking <tracking>] [-ownerNode <positive_integer>]
```

Arguments

id

Integer value that uniquely identifies the VMAC address. The generic VMAC address is in the form of 00:00:5e:00:01:<VRID>. For example, if you add a VRID with a value of 60 and bind it to an interface, the resulting VMAC address is 00:00:5e:00:01:3c, where 3c is the hexadecimal representation of 60.

Minimum value: 1

Maximum value: 255

priority

Base priority (BP), in an active-active mode configuration, which ordinarily determines the master VIP address.

Default value: 255

Minimum value: 1

Maximum value: 255

preemption

In an active-active mode configuration, make a backup VIP address the master if its priority becomes higher than that of a master VIP address bound to this VMAC address.

If you disable pre-emption while a backup VIP address is the master, the backup VIP address remains master until the original master VIP's priority becomes higher than that of the current master.

Possible values: ENABLED, DISABLED

Default value: ENABLED

sharing

In an active-active mode configuration, enable the backup VIP address to process any traffic instead of dropping it.

Possible values: ENABLED, DISABLED

Default value: DISABLED

tracking

The effective priority (EP) value, relative to the base priority (BP) value in an active-active mode configuration. When EP is set to a value other than None, it is EP, not BP, which determines the master VIP address.

Available settings function as follows:

* NONE - No tracking. EP = BP

* ALL - If the status of all virtual servers is UP, EP = BP. Otherwise, EP = 0.

* ONE - If the status of at least one virtual server is UP, EP = BP. Otherwise, EP = 0.

* PROGRESSIVE - If the status of all virtual servers is UP, EP = BP. If the status of all virtual servers is DOWN, EP = 0. Otherwise EP = BP $(1 - K/N)$, where N is the total number of virtual servers associated with the VIP address and K is the number of virtual servers for which the status is DOWN.

Default: NONE.

Possible values: NONE, ONE, ALL, PROGRESSIVE

Default value: NONE

ownerNode

Assign a cluster node as the owner of this VMAC address. If no owner is configured, owner node is displayed as ALL and one node is dynamically elected as the owner.

Default value: -1

Minimum value: 0

Maximum value: 31

Example

```
set vrID 1 -priority 100
```

unset vrID

Use this command to remove vrID settings. Refer to the set vrID command for meanings of the arguments.

Synopsys

```
unset vrID <id> [-priority] [-preemption] [-sharing] [-tracking] [-ownerNode]
```

bind vrID

Binds the specified interfaces to a VMAC configuration.

Synopsys

```
bind vrID <id> -ifnum <interface_name> ...
```

Arguments

id

Integer that uniquely identifies the VMAC address. The generic VMAC address is in the form of 00:00:5e:00:01:<VRID>. For example, if you add a VRID with a value of 60 and bind it to an interface, the resulting VMAC address is 00:00:5e:00:01:3c, where 3c is the hexadecimal representation of 60.

Minimum value: 1

Maximum value: 255

ifnum

Interfaces to bind to the VMAC, specified in (slot/port) notation (for example, 1/2). Use spaces to separate multiple entries.

Example

```
add vrID 1
```

unbind vrID

Unbinds specified interfaces from a VMAC configuration.

Synopsys

```
unbind vrID <id> -ifnum <interface_name> ...
```

Arguments

id

Integer value that uniquely identifies the VMAC address. The generic VMAC address is in the form of 00:00:5e:00:01:<VRID>. For example, if you add a VRID with a value of 60 and bind it to an interface, the resulting VMAC address is 00:00:5e:00:01:3c, where 3c is the hexadecimal representation of 60.

Minimum value: 1

Maximum value: 255

ifnum

Interfaces to unbind from the VMAC, specified in (slot/port) notation (for example, 1/2). Use spaces to separate multiple entries.

show vrid

Displays the settings of all VRIDs configured on the NetScaler appliance, or of the specified VRID. To display the settings of all the VRIDs, run the command without any parameters. To display the settings of a particular VRID, specify the VRID.

Synopsys

show vrid [<id>]

Arguments

id

Integer value that uniquely identifies the VMAC address.

Minimum value: 1

Maximum value: 255

Outputs

ifaces

Interfaces bound to this VRID.

type

Indicates whether this VRID entry was added manually or dynamically. When you manually add a VRID entry, the value for this parameter is STATIC. Otherwise, it is DYNAMIC.

vlan

The VLAN in which this VRID resides.

priority

Base priority (BP), in an active-active mode configuration, which ordinarily determines the master VIP address.

effectivePriority

The effective priority of this VRID.

preemption

In an active-active mode configuration, make a backup VIP address the master if its priority becomes higher than that of a master VIP address bound to this VMAC address.

If you disable pre-emption while a backup VIP address is the master, the backup VIP address remains master until the original master VIP's priority becomes higher than that of the current master.

sharing

In an active-active mode configuration, enable the backup VIP address to process any traffic instead of dropping it.

tracking

The effective priority (EP) value, relative to the base priority (BP) value in an active-active mode configuration. When EP is set to a value other than None, it is EP, not BP, which determines the master VIP address.

Available settings function as follows:

- * NONE - No tracking. EP = BP

- * ALL - If the status of all virtual servers is UP, EP = BP. Otherwise, EP = 0.

- * ONE - If the status of at least one virtual server is UP, EP = BP. Otherwise, EP = 0.

- * PROGRESSIVE - If the status of all virtual servers is UP, EP = BP. If the status of all virtual servers is DOWN, EP = 0. Otherwise EP = BP $(1 - K/N)$, where N is the total number of virtual servers associated with the VIP address and K is the number of virtual servers for which the status is DOWN.

Default: NONE.

flags

Flags.

IPAddress

The IP address bound to the VRID.

state

State of this VRID.

stateflag

operationalOwnerNode

Run time owner node of the vrid.

ownerNode

Assign a cluster node as the owner of this VMAC address. If no owner is configured, owner node is displayed as ALL and one node is dynamically elected as the owner.

devno

count

Example

```
show vrid
```

vrID6

The following operations can be performed on "vrID6":

[add](#) | [rm](#) | [bind](#) | [unbind](#) | [show](#)

add vrID6

Adds a VMAC6 address to the NetScaler appliance. A Virtual MAC address (VMAC6) is a floating entity, shared by the nodes in an HA configuration.

Synopsys

```
add vrID6 <id>
```

Arguments

id

Integer value that uniquely identifies a VMAC6 address.

Minimum value: 1

Maximum value: 255

Example

```
add vrID6 1
```

rm vrID6

Removes a specified VMAC6 entry or all VMAC6 entries from the NetScaler appliance.

Synopsys

```
rm vrID6 (<id> | -all)
```

Arguments

id

Integer value that uniquely identifies a VMAC6 address.

Minimum value: 1

Maximum value: 255

all

Remove all configured VMAC6 addresses from the NetScaler appliance.

bind vrID6

Binds the specified interfaces to a VMAC6 configuration.

Synopsys

```
bind vrID6 <id> -ifnum <interface_name> ...
```

Arguments

id

Integer value that uniquely identifies a VMAC6 address.

Minimum value: 1

Maximum value: 255

ifnum

Interfaces to bind to the VMAC6, specified in (slot/port) notation (for example, 1/2). Use spaces to separate multiple entries.

Example

```
add vrid6 1
```

unbind vrid6

Unbinds the specified interfaces from a VMAC6 configuration.

Synopsis

```
unbind vrid6 <id> -ifnum <interface_name> ...
```

Arguments

id

Integer value that uniquely identifies a VMAC6 address.

Minimum value: 1

Maximum value: 255

ifnum

Interfaces to unbind from the VMAC6, specified in (slot/port) notation (for example, 1/2). Use spaces to separate multiple entries.

show vrid6

Displays the settings of all VRID6s configured on the NetScaler appliance, or of the specified VRID6. To display the settings of all the VRID6s, run the command without any parameters. To display the settings of a particular VRID6, specify the VRID6.

Synopsis

```
show vrid6 [<id>]
```

Arguments

id

Integer value that uniquely identifies a VMAC6 address.

Minimum value: 1

Maximum value: 255

Outputs

ifaces

Interfaces bound to this VRID.

ifnum

Interfaces bound to this vrid.

type

Type (static or dynamic) of this VRID.

vlan

The VLAN in which this VRID resides.

priority

The priority of this VRID.

state

State of this VRID.

flags

Flags.

stateflag

IPAddress

The IP address bound to the VRID6

devno

count

Example

```
show vrid6
```


vrIDParam

The following operations can be performed on "vrIDParam":

[set](#) | [unset](#) | [show](#)

set vrIDParam

Sets global parameters of VMACs on the NetScaler appliance.

Synopsys

```
set vrIDParam -sendToMaster ( ENABLED | DISABLED )
```

Arguments

sendToMaster

Forward packets to the master node, in an active-active mode configuration, if the virtual server is in the backup state and sharing is disabled.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
set vrIDParam -sendToMaster ENABLED
```

unset vrIDParam

Use this command to remove vrIDParam settings. Refer to the set vrIDParam command for meanings of the arguments.

Synopsys

```
unset vrIDParam -sendToMaster
```

show vrIDParam

Displays the VRID global settings on the NetScaler appliance.

Synopsys

```
show vrIDParam
```

Outputs

sendToMaster

Forward packets to the master node, in an active-active mode configuration, if the virtual server is in the backup state and sharing is disabled.

vxlan

The following operations can be performed on "vxlan":

add | **rm** | **set** | **unset** | **bind** | **unbind** | **show** | **stat**

add vxlan

Adds a VXLAN to the NetScaler appliance.

Synopsys

```
add vxlan <id> [-vlan <positive_integer>] [-port <port>]
```

Arguments

id

A positive integer, which is also called VXLAN Network Identifier (VNI), that uniquely identifies a VXLAN.

Minimum value: 1

Maximum value: 16777215

vlan

ID of VLANs whose traffic is allowed over this VXLAN. If you do not specify any VLAN IDs, the NetScaler allows traffic of all VLANs that are not part of any other VXLANs.

Minimum value: 1

Maximum value: 4094

port

Specifies UDP destination port for VXLAN packets.

Default value: 4789

Minimum value: 1

Maximum value: 65534

Example

```
add vxlan 20000 -vlan 4
```

rm vxlan

Removes a VXLAN from the NetScaler appliance

Synopsys

```
rm vxlan <id>
```

Arguments

id

A positive integer, which is also called VXLAN Network Identifier (VNI), that uniquely identifies a VXLAN.

Minimum value: 1

Maximum value: 16777215

Example

```
rm vxlan 20000
```

set vxlan

Modify VXLAN parameters

Synopsys

```
set vxlan <id> [-vlan <positive_integer>] [-port <port>]
```

Arguments

id

A positive integer, which is also called VXLAN Network Identifier (VNI), that uniquely identifies a VXLAN.

Minimum value: 1

Maximum value: 16777215

vlan

ID of VLANs whose traffic is allowed over this VXLAN. If you do not specify any VLAN IDs, the NetScaler allows traffic of all VLANs that are not part of any other VXLANs.

Minimum value: 1

Maximum value: 4094

port

Specifies UDP destination port for VXLAN packets.

Default value: 4789

Minimum value: 1

Maximum value: 65534

Example

```
set vxlan 20000 -vlan 4
```

unset vxlan

Use this command to remove vxlan settings. Refer to the set vxlan command for meanings of the arguments.

Synopsys

```
unset vxlan <id> [-vlan] [-port]
```

bind vxlan

Binds tunnels or IP addresses to the VXLAN

Synopsys

```
bind vxlan <id> (-tunnel <string> | (-IPAddress <ip_addr|ipv6_addr|*> [<netmask>]))
```

Arguments

id

A positive integer, which is also called VXLAN Network Identifier (VNI), that uniquely identifies a VXLAN.

Minimum value: 1

Maximum value: 16777215

tunnel

Specifies the name of the configured tunnel to be associated with this VXLAN.

IPAddress

Network address to be associated with the VXLAN. Should exist on the appliance before you associate it with the VXLAN.

netmask

Subnet mask for the network address defined for this VXLAN.

Example

```
bind vxlan 20000 -tunnel t1
```

unbind vxlan

Unbinds tunnels and IP addresses from the VXLAN

Synopsys

```
unbind vxlan <id> (-tunnel <string> | (-IPAddress <ip_addr|ipv6_addr|*> [<netmask>]))
```

Arguments

id

A positive integer, which is also called VXLAN Network Identifier (VNI), that uniquely identifies a VXLAN.

Minimum value: 1

Maximum value: 16777215

tunnel

Specifies the name of the configured tunnel to be associated with this VXLAN.

IPAddress

The IP Address associated with the VXLAN configuration.

netmask

Subnet mask for the network address defined for this VXLAN.

Example

```
unbind vxlan 20000 -tunnel t1
```

show vxlan

Display all the VXLANs on the Netscaler appliance

Synopsys

```
show vxlan [<id>]
```

Arguments

id

A positive integer, which is also called VXLAN Network Identifier (VNI), that uniquely identifies a VXLAN.

Minimum value: 1

Maximum value: 16777215

Outputs

vlan

ID of VLANs whose traffic is allowed over this VXLAN. If you do not specify any VLAN IDs, the NetScaler allows traffic of all VLANs that are not part of any other VXLANs.

port

Specifies UDP destination port for VXLAN packets.

tunnel

Specifies the name of the configured tunnel to be associated with this VXLAN.

IPAddress

The IP address assigned to the VXLAN.

netmask

Subnet mask for the network address defined for this VXLAN.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

stateflag

Internal flag for display

devno

count

stat vxlan

Display statistics for VXLAN(s).

Synopsys

```
stat vxlan [<id>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

id

An integer specifying the VXLAN identification number (VNID).

Minimum value: 1

Maximum value: 16777215

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Packets received (RxPkts)

Packets received on the VXLAN.

Bytes received (RxBytes)

Bytes of data received on the VXLAN.

Packets sent (TxPkts)

Packets transmitted on the VXLAN.

Bytes sent (TxBytes)

Bytes of data transmitted on the VXLAN.

Example

```
stat vxlan 10000
```

NS Commands

The entities on which you can perform NetScaler CLI operations:

- o ns
- o ns acl
- o ns acl6
- o ns acls
- o ns acls6
- o ns appflowCollector
- o ns appflowParam
- o ns aptlicense
- o ns assignment
- o ns config
- o ns connectiontable
- o ns consoleloginprompt
- o ns dhcpIp
- o ns dhcpParams
- o ns diameter
- o ns encryptionParams
- o ns events
- o ns feature
- o ns hardware
- o ns hostName
- o ns httpParam
- o ns httpProfile
- o ns idletimeout
- o ns info
- o ns ip
- o ns ip6
- o ns license
- o ns limitIdentifier
- o ns limitSelector
- o ns limitSessions
- o ns memory
- o ns mode
- o ns ns.conf
- o ns param
- o ns pbr
- o ns pbr6
- o ns pbrs
- o ns persistencesession
- o ns rateControl
- o ns rollbackcmd
- o ns rpcNode
- o ns runningConfig
- o ns savedConfig
- o ns simpleacl
- o ns simpleacl6
- o ns spParams
- o ns stats
- o ns surgeQ
- o ns tcpParam
- o ns tcpProfile
- o ns tcpbufParam
- o ns timeout
- o ns timer
- o ns trafficDomain
- o ns variable
- o ns version
- o ns weblogparam
- o ns xmlnsnamespace

- [reboot](#)
- [shutdown](#)

ns

The following operations can be performed on "ns":

[config](#) | [stat](#)

config ns

Displays a menu to configure the basic parameters of a NetScaler appliance. Note: The appliance must be rebooted for these changes to take effect.

Synopsys

config ns

stat ns

Displays generic statistics of the NetScaler appliance.

Synopsys

stat ns [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

Maximum Memory Available (MB) (MemTotAvail)

Total system memory available for PE to grab from the system.

Average CPU usage (%) (CPU)

Average CPU utilization percentage. Not applicable for a single-CPU system.

CPU usage (%) (CPU)

CPU utilization percentage.

Maximum memory(KB) Deprecated (MaxMem)

Largest amount of memory the NetScaler can dedicate to caching, up to 50% of available memory. A 0 value disables caching, but the caching module continues to run.

Delta compression ratio (DICmpRt)

Ratio of compressible data received to compressed data transmitted. If this ratio is one (uncmp:1.0) that means compression is disabled or we are not able to compress even a single compressible packet.

Average CPU usage (CPU)

Shows average CPU utilization percentage if more than 1 CPU is present.

CPU usage (CPU)

CPU utilization: percentage * 10.

Memory usage (MemUsage)

Percentage of memory utilization on NetScaler.

Total HTTP compression ratio

Ratio of total HTTP data received to total HTTP data transmitted.

HTTP compression ratio

Ratio of the compressible data received from the server to the compressed data sent to the client.

Utilized memory(KB) (UtiMem)

Amount of memory the integrated cache is currently using.

Maximum memory active value(KB) (MaxMemActive)

Currently active value of maximum memory

Maximum memory(KB) (Max64Mem)

Largest amount of memory the NetScaler can dedicate to caching, up to 50% of available memory. A 0 value disables caching, but the caching module continues to run.

Origin bandwidth saved(%) (POrBan)

Percentage of origin bandwidth saved, expressed as number of bytes served from the integrated cache divided by all bytes served. The assumption is that all compression is done in the NetScaler.

Misses (TotMiss)

Intercepted HTTP requests requiring fetches from origin server.

Hits (TotHit)

Responses served from the integrated cache. These responses match a policy with a CACHE action.

SSL cards UP (SSLCardUP)

Number of SSL cards that are UP. If the number of cards UP is lower than a threshold, a failover is initiated.

InUse Memory (%) (MemUsage)

Percentage of memory utilization on NetScaler.

Memory usage (MB) (MemUseMB)

Main memory currently in use, in megabytes.

Management CPU usage (%) (CPU)

Management CPU utilization percentage.

Packet CPU usage (%) (CPU)

Average CPU utilization percentage for all packet engines excluding management PE.

Up since (Since)

Time when the NetScaler appliance was last started.

Last Transition time (TransTime)

Time when the last master state transition occurred. You can use this statistic for debugging.

System state (HAState)

State of the HA node, based on its health, in a high availability setup. Possible values are:

UP ? Indicates that the node is accessible and can function as either a primary or secondary node.

DISABLED ? Indicates that the high availability status of the node has been manually disabled. Synchronization and propagation cannot take place between the peer nodes.

INIT ? Indicates that the node is in the process of becoming part of the high availability configuration.

PARTIALFAIL ? Indicates that one of the high availability monitored interfaces has failed because of a card or link failure. This state triggers a failover.

COMPLETEFAIL ? Indicates that all the interfaces of the node are unusable, because the interfaces on which high availability monitoring is enabled are not connected or are manually disabled. This state triggers a failover.

DUMB ? Indicates that the node is in listening mode. It does not participate in high availability transitions or transfer configuration from the peer node. This is a configured value, not a statistic.

PARTIALFAILSSL ? Indicates that the SSL card has failed. This state triggers a failover.

ROUTEMONITORFAIL ? Indicates that the route monitor has failed. This state triggers a failover.

Master state (mastate)

Indicates the high availability state of the node. Possible values are:

STAYSECONDARY ? Indicates that the selected node remains the secondary node in a high availability setup. In this case a forced failover does not change the state but, instead, returns an appropriate error message. This is a configured value and not a statistic.

PRIMARY ? Indicates that the selected node is the primary node in a high availability setup.

SECONDARY ? Indicates that the selected node is the secondary node in a high availability setup.

CLAIMING ? Indicates that the secondary node is in the process of taking over as the primary node. This is the intermediate state in the transition of the secondary node to primary status.

FORCE CHANGE - Indicates that the secondary node is forcibly changing its status to primary due to a forced failover issued on the secondary node.

SSL cards present (SSLCards)

Number of SSL crypto cards present on the NetScaler appliance.

/flash Used (%) (disk0PerUsage)

Used space in /flash partition of the disk, as a percentage. This is a critical counter.

You can configure /flash Used (%) by using the Set snmp alarm DISK-USAGE-HIGH command.

/var Used (%) (disk1PerUsage)

Used space in /var partition of the disk, as a percentage. This is a critical counter. You can configure /var Used (%) by using the Set snmp alarm DISK-USAGE-HIGH command.

/flash Available (MB) (disk0Avail)

Available space in /flash partition of the hard disk.

/var Available (MB) (disk1Avail)

Available space in /var partition of the hard disk.

Megabits received (RxMb)

Number of megabytes received by the NetScaler appliance.

Megabits transmitted (TxMb)

Number of megabytes transmitted by the NetScaler appliance.

All client connections (CltCx)

Client connections, including connections in the Opening, Established, and Closing state.

Established client connections (CltCxE)

Current client connections in the Established state, which indicates that data transfer can occur between the NetScaler and the client.

All server connections (SvrCx)

Server connections, including connections in the Opening, Established, and Closing state.

Established server connections (SvrCxE)

Current server connections in the Established state, which indicates that data transfer can occur between the NetScaler and the server.

Total requests (HTReqRx)

Total number of HTTP requests received.

Total responses (HTRspRx)

Total number of HTTP responses sent.

Request bytes received (HTReqbRx)

Total number of bytes of HTTP request data received.

Response bytes received (HTRspbRx)

Total number of bytes of HTTP response data received.

SSL transactions (SSLTrn)

Number of SSL transactions on the NetScaler appliance.

SSL session hits (SeHit)

Number of SSL session reuse hits on the NetScaler appliance.

requests (reqs)

HTTP/HTTPS requests sent to your protected web servers via the Application Firewall.

responses (resps)

HTTP/HTTPS responses sent by your protected web servers via the Application Firewall.

aborts

Incomplete HTTP/HTTPS requests aborted by the client before the Application Firewall could finish processing them.

redirects (redirect)

HTTP/HTTPS requests redirected by the Application Firewall to a different Web page or web server. (HTTP 302)

Misc. Counter 0 (misc0)

Miscellaneous Counter 0.

Misc. Counter 1 (misc1)

Miscellaneous Counter 1.

Management CPU usage (CPU)

Management CPU utilization: percentage * 10.

SSL crypto card status (SSLCardSt)

Status of the SSL card(s). The value should be interpreted in binary form, with each set bit indicates a card as UP.

304 hits (304Hit)

Object not modified responses served from the cache. (Status code 304 served instead of the full response.)

Non-304 hits (Non304Hit)

Total number of full (non-304) responses served from the cache. A 304 status code indicates that a response has not been modified since the last time it was served

sql hits (sqlHit)

sql response served from cache

Requests (CacReq)

Total cache hits plus total cache misses.

Compressed bytes transmitted

Number of bytes the NetScaler sends to the client after compressing the response from the server.

Compressible bytes received

Number of bytes that can be compressed, which the NetScaler receives from the server. This gives the content length of the response that the NetScaler receives from server.

Compressible bytes received (DICmpRxB)

Total number of delta-compressible bytes received by NetScaler.

Compressed bytes transmitted (DICmpTxB)

Total number of delta-compressed bytes transmitted by NetScaler.

ns acl

The following operations can be performed on "ns acl":

add | **rm** | **set** | **unset** | **enable** | **disable** | **stat** | **rename** | **show**

add ns acl

Adds an extended ACL rule to the NetScaler appliance. To commit this operation, you must apply the extended ACLs. Extended ACL rules filter data packets on the basis of various parameters, such as IP address, source port, action, and protocol.

Synopsys

```
add ns acl <aclname> <aclaction> [-td <positive_integer>] [-srcIP [<operator>] <srcIPVal>] [-srcPort [<operator>] <srcPortVal>] [-destIP [<operator>] <destIPVal>] [-destPort [<operator>] <destPortVal>] [-TTL <positive_integer>] [-srcMac <mac_addr>] [(-protocol <protocol> [-established]) | -protocolNumber <positive_integer>] [-vlan <positive_integer>] [-vxlan <positive_integer>] [-interface <interface_name>] [-icmpType <positive_integer>] [-icmpCode <positive_integer>]] [-priority <positive_integer>] [-state ( ENABLED | DISABLED )] [-logstate ( ENABLED | DISABLED )] [-ratelimit <positive_integer>]]
```

Arguments

aclname

Name for the extended ACL rule. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Can be changed after the extended ACL rule is created.

aclaction

Action to perform on incoming IPv4 packets that match the extended ACL rule.

Available settings function as follows:

- * ALLOW - The NetScaler appliance processes the packet.
- * BRIDGE - The NetScaler appliance bridges the packet to the destination without processing it.
- * DENY - The NetScaler appliance drops the packet.

Possible values: BRIDGE, DENY, ALLOW

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

srcIP

IP address or range of IP addresses to match against the source IP address of an incoming IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [10.102.29.30-10.102.29.189].

operator

Either the equals (=) or does not equal (!=) logical operator.

Possible values: =, !=, EQ, NEQ

srcIPVal

IP address or range of IP addresses to match against the source IP address of an incoming IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [10.102.29.30-10.102.29.189].

srcPort

Port number or range of port numbers to match against the source port number of an incoming IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

srcPortVal

Port number or range of port numbers to match against the source port number of an incoming IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

Maximum value: 65535

destIP

IP address or range of IP addresses to match against the destination IP address of an incoming IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [10.102.29.30-10.102.29.189].

destIPVal

IP address or range of IP addresses to match against the destination IP address of an incoming IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [10.102.29.30-10.102.29.189].

destPort

Port number or range of port numbers to match against the destination port number of an incoming IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

Note: The destination port can be specified only for TCP and UDP protocols.

destPortVal

Port number or range of port numbers to match against the destination port number of an incoming IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

Note: The destination port can be specified only for TCP and UDP protocols.

Maximum value: 65535

TTL

Number of seconds, in multiples of four, after which the extended ACL rule expires. If you do not want the extended ACL rule to expire, do not specify a TTL value.

Minimum value: 1

Maximum value: 2147483647

srcMac

MAC address to match against the source MAC address of an incoming IPv4 packet.

protocol

Protocol to match against the protocol of an incoming IPv4 packet.

Possible values: ICMP, IGMP, TCP, EGP, IGP, ARGUS, UDP, RDP, RSVP, EIGRP, L2TP, ISIS

protocolNumber

Protocol to match against the protocol of an incoming IPv4 packet.

Minimum value: 1

Maximum value: 255

vlan

ID of the VLAN. The NetScaler appliance applies the ACL rule only to the incoming packets of the specified VLAN. If you do not specify a VLAN ID, the appliance applies the ACL rule to the incoming packets on all VLANs.

Minimum value: 1

Maximum value: 4094

vxlan

ID of the VXLAN. The NetScaler appliance applies the ACL rule only to the incoming packets of the specified VXLAN. If you do not specify a VXLAN ID, the appliance applies the ACL rule to the incoming packets on all VXLANs.

Minimum value: 1

Maximum value: 16777215

interface

ID of an interface. The NetScaler appliance applies the ACL rule only to the incoming packets from the specified interface. If you do not specify any value, the appliance applies the ACL rule to the incoming packets of all interfaces.

established

Allow only incoming TCP packets that have the ACK or RST bit set, if the action set for the ACL rule is ALLOW and these packets match the other conditions in the ACL rule.

icmpType

ICMP Message type to match against the message type of an incoming ICMP packet. For example, to block DESTINATION UNREACHABLE messages, you must specify 3 as the ICMP type.

Note: This parameter can be specified only for the ICMP protocol.

Minimum value: 0

Maximum value: 65536

icmpCode

Code of a particular ICMP message type to match against the ICMP code of an incoming ICMP packet. For example, to block DESTINATION HOST UNREACHABLE messages, specify 3 as the ICMP type and 1 as the ICMP code.

If you set this parameter, you must set the ICMP Type parameter.

Minimum value: 0

Maximum value: 65536

priority

Priority for the extended ACL rule that determines the order in which it is evaluated relative to the other extended ACL rules. If you do not specify priorities while creating extended ACL rules, the ACL rules are evaluated in the order in which they are created.

Minimum value: 1

Maximum value: 100000

state

Enable or disable the extended ACL rule. After you apply the extended ACL rules, the NetScaler appliance compares incoming packets against the enabled extended ACL rules.

Possible values: ENABLED, DISABLED

Default value: ENABLED

logstate

Enable or disable logging of events related to the extended ACL rule. The log messages are stored in the configured syslog or auditlog server.

Possible values: ENABLED, DISABLED

Default value: DISABLED

ratelimit

Maximum number of log messages to be generated per second. If you set this parameter, you must enable the Log State parameter.

Default value: 100

Minimum value: 1

Maximum value: 10000

Example

```
add ns acl restrict DENY -srcport 45-1024 -destIP 192.168.1.1 -protocol TCP
```

rm ns acl

Removes an extended ACL rule from the NetScaler appliance. To commit this operation, you must apply the extended ACLs.

Synopsys

```
rm ns acl <aclname> ...
```

Arguments

aclname

Name of the extended ACL rule that you want to remove.

Example

```
rm ns acl restrict
```

set ns acl

Modifies the parameters of an ACL rule. To commit this operation, you must apply the extended ACLs.

Synopsys

```
set ns acl <aclname> [-aclaction <aclaction>] [-srcIP [<operator>] <srcIPVal>] [-srcPort [<operator>] <srcPortVal>] [-destIP [<operator>] <destIPVal>] [-destPort [<operator>] <destPortVal>] [-srcMac <mac_addr>] [-protocol <protocol>] | -protocolNumber <positive_integer>] [-icmpType <positive_integer>] [-icmpCode <positive_integer>]] [-vlan <positive_integer>] [-vxlan <positive_integer>] [-interface <interface_name>] [-priority <positive_integer>] [-logstate (ENABLED | DISABLED)] [-ratelimit <positive_integer>] [-established]
```

Arguments

aclname

Name of the ACL rule whose parameters you want to modify.

aclaction

Action to perform on incoming IPv4 packets that match the extended ACL rule.

Available settings function as follows:

- * ALLOW - The NetScaler appliance processes the packet.
- * BRIDGE - The NetScaler appliance bridges the packet to the destination without processing it.
- * DENY - The NetScaler appliance drops the packet.

Possible values: BRIDGE, DENY, ALLOW

srcIP

IP address or range of IP addresses to match against the source IP address of an incoming IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [10.102.29.30-10.102.29.189].

operator

Either the equals (=) or does not equal (!=) logical operator.

Possible values: =, !=, EQ, NEQ

srcIPVal

IP address or range of IP addresses to match against the source IP address of an incoming IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [10.102.29.30-10.102.29.189].

srcPort

Port number or range of port numbers to match against the source port number of an incoming IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

srcPortVal

Port number or range of port numbers to match against the source port number of an incoming IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

Maximum value: 65535

destIP

IP address or range of IP addresses to match against the destination IP address of an incoming IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [10.102.29.30-10.102.29.189].

destIPVal

IP address or range of IP addresses to match against the destination IP address of an incoming IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [10.102.29.30-10.102.29.189].

destPort

Port number or range of port numbers to match against the destination port number of an incoming IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

Note: The destination port can be specified only for TCP and UDP protocols.

destPortVal

Port number or range of port numbers to match against the destination port number of an incoming IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

Note: The destination port can be specified only for TCP and UDP protocols.

Maximum value: 65535

srcMac

MAC address to match against the source MAC address of an incoming IPv4 packet.

protocol

Protocol to match against the protocol of an incoming IPv4 packet.

Possible values: ICMP, IGMP, TCP, EGP, IGP, ARGUS, UDP, RDP, RSVP, EIGRP, L2TP, ISIS

protocolNumber

Protocol to match against the protocol of an incoming IPv4 packet.

Minimum value: 1

Maximum value: 255

icmpType

ICMP Message type to match against the message type of an incoming ICMP packet. For example, to block DESTINATION UNREACHABLE messages, you must specify 3 as the ICMP type.

Note: This parameter can be specified only for the ICMP protocol.

Minimum value: 0

Maximum value: 65536

icmpCode

Code of a particular ICMP message type to match against the ICMP code of an incoming ICMP packet. For example, to block DESTINATION HOST UNREACHABLE messages, specify 3 as the ICMP type and 1 as the ICMP code.

If you set this parameter, you must set the ICMP Type parameter.

Minimum value: 0

Maximum value: 65536

vlan

ID of the VLAN. The NetScaler appliance applies the ACL rule only to the incoming packets of the specified VLAN. If you do not specify a VLAN ID, the appliance applies the ACL rule to the incoming packets on all VLANs.

Minimum value: 1

Maximum value: 4094

vxlan

ID of the VXLAN. The NetScaler appliance applies the ACL rule only to the incoming packets of the specified VXLAN. If you do not specify a VXLAN ID, the appliance applies the ACL rule to the incoming packets on all VXLANs.

Minimum value: 1

Maximum value: 16777215

interface

ID of an interface. The NetScaler appliance applies the ACL rule only to the incoming packets from the specified interface. If you do not specify any value, the appliance applies the ACL rule to the incoming packets of all interfaces.

priority

Priority for the extended ACL rule that determines the order in which it is evaluated relative to the other extended ACL rules. If you do not specify priorities while creating extended ACL rules, the ACL rules are evaluated in the order in which they are created.

Minimum value: 1

Maximum value: 100000

logstate

Enable or disable logging of events related to the extended ACL rule. The log messages are stored in the configured syslog or auditlog server.

Possible values: ENABLED, DISABLED

Default value: DISABLED

ratelimit

Maximum number of log messages to be generated per second. If you set this parameter, you must enable the Log State parameter.

Default value: 100

Minimum value: 1

Maximum value: 10000

established

Allow only incoming TCP packets that have the ACK or RST bit set, if the action set for the ACL rule is ALLOW and these packets match the other conditions in the ACL rule.

Example

```
set ns acl restrict -srcPort 50
```

unset ns acl

Resets the attributes of the specified extended ACL rule. Attributes for which a default value is available revert to their default values. Refer to the set ns acl command for a description of the parameters..Refer to the set ns acl command for meanings of the arguments.

Synopsys

```
unset ns acl <aclname> [-srcIP] [-srcPort] [-destIP] [-destPort] [-srcMac] [-protocol] [-icmpType] [-icmpCode] [-vlan] [-vxlan] [-interface] [-logstate] [-ratelimit] [-established]
```

Example

```
unset ns acl rule1 -srcPort
```

enable ns acl

Enables an extended ACL rule. To commit this operation, you must apply the extended ACLs. After you apply the extended ACL rules, the NetScaler appliance compares incoming packets against the enabled extended ACL rules.

Synopsys

```
enable ns acl <aclname> ...
```

Arguments

aclname

Name of the extended ACL rule that you want to enable.

Example

```
enable ns acl foo
```

disable ns acl

Disables an extended ACL rule. To commit this operation, you must apply the extended ACLs. After you apply the ACL rules, the NetScaler appliance does not compare incoming packets against the disabled extended ACL rules.

Synopsys

```
disable ns acl <aclname> ...
```

Arguments

aclname

Name of the extended ACL rule that you want to disable.

Example

```
disable ns acl foo
```

stat ns acl

Displays statistics related to the extended ACL rules. To display statistics of all the extended ACL rules, run the command without any parameters. To display statistics of a particular extended ACL rule, specify the name of the extended ACL rule.

Synopsys

```
stat ns acl [<aclname>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full )]
```

Arguments

aclname

Name of the extended ACL rule whose statistics you want the NetScaler appliance to display.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Bridge ACL hits (ACLBdg)

Packets matching a bridge ACL, which is in transparent mode and bypasses service processing.

Deny ACL hits (ACLDeny)

Packets dropped because they match ACLs with processing mode set to DENY.

Allow ACL hits (ACLAllow)

Packets matching ACLs with processing mode set to ALLOW. NetScaler processes these packets.

NAT ACL hits (ACLNAT)

Packets matching a NAT ACL, resulting in a NAT session.

ACL hits (ACLSHits)

Packets matching an ACL.

ACL misses (ACLMiss)

Packets not matching any ACL.

ACL Count (ACLCount)

Total number of ACL rules configured.

Hits for this ACL (Hits)

Number of times the acl was hit

Example

```
stat acl
```

rename ns acl

Renames an extended ACL rule.

Synopsys

```
rename ns acl <aclname> <newName>
```

Arguments

aclname

Name of the extended ACL rule that you want to rename.

newName

New name for the extended ACL rule. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

Example

```
rename acl rule rule-new
```

show ns acl

Displays settings related to the extended ACL rules. To display settings of all the extended ACL rules, run the command without any parameters. To display settings of a particular extended ACL rule, specify the name of the extended ACL rule.

Synopsys

```
show ns acl [<aclname>]
```

Arguments

aclname

Name of the extended ACL rule whose details you want the NetScaler appliance to display.

Outputs

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

aclaction

Action to perform on incoming IPv4 packets that match the extended ACL rule.

Available settings function as follows:

- * ALLOW - The NetScaler appliance processes the packet.
- * BRIDGE - The NetScaler appliance bridges the packet to the destination without processing it.
- * DENY - The NetScaler appliance drops the packet.

srcMac

MAC address to match against the source MAC address of an incoming IPv4 packet.

stateflag

ACL state flag.

protocol

The protocol number in IP header or name.

protocolNumber

The protocol number in IP header or name.

srcPortVal

Port number or range of port numbers to match against the source port number of an incoming IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

destPortVal

Port number or range of port numbers to match against the destination port number of an incoming IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

Note: The destination port can be specified only for TCP and UDP protocols.

srcIPVal

IP address or range of IP addresses to match against the source IP address of an incoming IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [10.102.29.30-10.102.29.189].

destIPVal

IP address or range of IP addresses to match against the destination IP address of an incoming IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [10.102.29.30-10.102.29.189].

vlan

ID of the VLAN. The NetScaler appliance applies the ACL rule only to the incoming packets of the specified VLAN. If you do not specify a VLAN ID, the appliance applies the ACL rule to the incoming packets on all VLANs.

vxlan

ID of the VXLAN. The NetScaler appliance applies the ACL rule only to the incoming packets of the specified VXLAN. If you do not specify a VXLAN ID, the appliance applies the ACL rule to the incoming packets on all VXLANs.

state

Enable or disable the extended ACL rule. After you apply the extended ACL rules, the NetScaler appliance compares incoming packets against the enabled extended ACL rules.

TTL

Number of seconds, in multiples of four, after which the extended ACL rule expires. If you do not want the extended ACL rule to expire, do not specify a TTL value.

icmpType

ICMP Message type to match against the message type of an incoming ICMP packet. For example, to block DESTINATION UNREACHABLE messages, you must specify 3 as the ICMP type.

Note: This parameter can be specified only for the ICMP protocol.

icmpCode

Code of a particular ICMP message type to match against the ICMP code of an incoming ICMP packet. For example, to block DESTINATION HOST UNREACHABLE messages, specify 3 as the ICMP type and 1 as the ICMP code.

If you set this parameter, you must set the ICMP Type parameter.

interface

ID of an interface. The NetScaler appliance applies the ACL rule only to the incoming packets from the specified interface. If you do not specify any value, the appliance applies the ACL rule to the incoming packets of all interfaces.

hits

The hits of this ACL.

established

This flag indicates that the ACL should be used for TCP response traffic only.

priority

Priority for the extended ACL rule that determines the order in which it is evaluated relative to the other extended ACL rules. If you do not specify priorities while creating extended ACL rules, the ACL rules are evaluated in the order in which they are created.

operator

Either the equals (=) or does not equal (!=) logical operator.

kernelstate

The commit status of the ACL.

logstate

Enable or disable logging of events related to the extended ACL rule. The log messages are stored in the configured syslog or auditlog server.

ratelimit

Packet rate limit for acl logging

time

Time when this acl is applied.

devno**count**

Example

```
sh acl foo      Name: foo      Action: ALLOW      Hits: 0      srcIP
```

ns acl6

The following operations can be performed on "ns acl6":

add | **rm** | **set** | **unset** | **enable** | **disable** | **stat** | **rename** | **show**

add ns acl6

Adds an ACL6 rule to the NetScaler appliance. To commit this operation, you must apply the ACL6s. ACL6 rules filter data packets on the basis of various parameters, such as IP address, source port, action, and protocol.

Synopsys

```
add ns acl6 <acl6name> <acl6action> [-td <positive_integer>] [-srcIPv6 [<operator>] <srcIPv6Val>] [-srcPort  
[<operator>] <srcPortVal>] [-destIPv6 [<operator>] <destIPv6Val>] [-destPort [<operator>] <destPortVal>] [-TTL  
<positive_integer>] [-srcMac <mac_addr>] [(-protocol <protocol> [-established]) | -protocolNumber  
<positive_integer>] [-vlan <positive_integer> | -vxlan <positive_integer>] [-interface <interface_name>] [-icmpType  
<positive_integer>] [-icmpCode <positive_integer>]] [-priority <positive_integer>] [-state ( ENABLED | DISABLED )]
```

Arguments

acl6name

Name for the ACL6 rule. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Can be changed after the ACL6 rule is created.

acl6action

Action to perform on the incoming IPv6 packets that match the ACL6 rule.

Available settings function as follows:

- * ALLOW - The NetScaler appliance processes the packet.
- * BRIDGE - The NetScaler appliance bridges the packet to the destination without processing it.
- * DENY - The NetScaler appliance drops the packet.

Possible values: BRIDGE, DENY, ALLOW

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

srcIPv6

IP address or range of IP addresses to match against the source IP address of an incoming IPv6 packet. In the command line interface, separate the range with a hyphen and enclose within brackets.

operator

Logical operator.

Possible values: =, !=, EQ, NEQ

srcIPv6Val

Source IPv6 address (range).

srcPort

Port number or range of port numbers to match against the source port number of an incoming IPv6 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

Note: The destination port can be specified only for TCP and UDP protocols.

srcPortVal

Source port (range).

Maximum value: 65535

destIPv6

IP address or range of IP addresses to match against the destination IP address of an incoming IPv6 packet. In the command line interface, separate the range with a hyphen and enclose within brackets.

destIPv6Val

Destination IPv6 address (range).

destPort

Port number or range of port numbers to match against the destination port number of an incoming IPv6 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

Note: The destination port can be specified only for TCP and UDP protocols.

destPortVal

Destination port (range).

Maximum value: 65535

TTL

Time to expire this ACL6 (in seconds).

Minimum value: 1

Maximum value: 2147483647

srcMac

MAC address to match against the source MAC address of an incoming IPv6 packet.

protocol

Protocol, identified by protocol name, to match against the protocol of an incoming IPv6 packet.

Possible values: ICMPV6, TCP, UDP

protocolNumber

Protocol, identified by protocol number, to match against the protocol of an incoming IPv6 packet.

Minimum value: 1

Maximum value: 255

vlan

ID of the VLAN. The NetScaler appliance applies the ACL6 rule only to the incoming packets on the specified VLAN. If you do not specify a VLAN ID, the appliance applies the ACL6 rule to the incoming packets on all VLANs.

Minimum value: 1

Maximum value: 4094

vxlan

ID of the VXLAN. The NetScaler appliance applies the ACL6 rule only to the incoming packets on the specified VXLAN. If you do not specify a VXLAN ID, the appliance applies the ACL6 rule to the incoming packets on all VXLANs.

Minimum value: 1

Maximum value: 16777215

interface

ID of an interface. The NetScaler appliance applies the ACL6 rule only to the incoming packets from the specified interface. If you do not specify any value, the appliance applies the ACL6 rule to the incoming packets from all interfaces.

established

Allow only incoming TCP packets that have the ACK or RST bit set if the action set for the ACL6 rule is ALLOW and these packets match the other conditions in the ACL6 rule.

icmpType

ICMP Message type to match against the message type of an incoming IPv6 ICMP packet. For example, to block DESTINATION UNREACHABLE messages, you must specify 3 as the ICMP type.

Note: This parameter can be specified only for the ICMP protocol.

Minimum value: 0

Maximum value: 65536

icmpCode

Code of a particular ICMP message type to match against the ICMP code of an incoming IPv6 ICMP packet. For example, to block DESTINATION HOST UNREACHABLE messages, specify 3 as the ICMP type and 1 as the ICMP code.

If you set this parameter, you must set the ICMP Type parameter.

Minimum value: 0

Maximum value: 65536

priority

Priority for the ACL6 rule, which determines the order in which it is evaluated relative to the other ACL6 rules. If you do not specify priorities while creating ACL6 rules, the ACL6 rules are evaluated in the order in which they are created.

Minimum value: 1

Maximum value: 80000

state

State of the ACL6.

Possible values: ENABLED, DISABLED

Default value: ENABLED

Example

```
add ns acl6 rule1 DENY -srcport 45-1024 -destIPv6 2001::45 -protocol TCP
```

rm ns acl6

Removes an ACL6 rule from the NetScaler appliance. To commit this operation, you must apply the ACL6s.

Synopsys

```
rm ns acl6 <acl6name> ...
```

Arguments

acl6name

Name of the ACL6 rule that you want to remove.

Example

```
rm ns acl6 rule1
```

set ns acl6

Modifies the parameters of an ACL6 rule. To commit this operation, you must apply the ACL6s.

Synopsys

```
set ns acl6 <acl6name> [-aclaction <aclaction>] [-srcIPv6 [<operator>] <srcIPv6Val>] [-srcPort [<operator>] <srcPortVal>] [-destIPv6 [<operator>] <destIPv6Val>] [-destPort [<operator>] <destPortVal>] [-srcMac <mac_addr>] [-protocol <protocol> | -protocolNumber <positive_integer>] [-icmpType <positive_integer>] [-icmpCode <positive_integer>]] [-vlan <positive_integer> | -vxlan <positive_integer>] [-interface <interface_name>] [-priority <positive_integer>] [-established]
```

Arguments

acl6name

Name of the ACL6 rule whose parameters you want to modify.

aclaction

Action associated with the ACL6.

Possible values: BRIDGE, DENY, ALLOW

srcIPv6

IP address or range of IP addresses to match against the source IP address of an incoming IPv6 packet. In the command line interface, separate the range with a hyphen and enclose within brackets.

operator

Logical operator.

Possible values: =, !=, EQ, NEQ

srcIPv6Val

Source IPv6 address (range).

srcPort

Source Port (range).

srcPortVal

Source port (range).

Maximum value: 65535

destIPv6

IP address or range of IP addresses to match against the destination IP address of an incoming IPv6 packet. In the command line interface, separate the range with a hyphen and enclose within brackets.

destIPv6Val

Destination IPv6 address (range).

destPort

Destination Port (range).

destPortVal

Destination port (range).

Maximum value: 65535

srcMac

MAC address to match against the source MAC address of an incoming IPv6 packet.

protocol

Protocol, identified by protocol name, to match against the protocol of an incoming IPv6 packet.

Possible values: ICMPV6, TCP, UDP

protocolNumber

Protocol, identified by protocol number, to match against the protocol of an incoming IPv6 packet.

Minimum value: 1

Maximum value: 255

icmpType

ICMP Message type to match against the message type of an incoming IPv6 ICMP packet. For example, to block DESTINATION UNREACHABLE messages, you must specify 3 as the ICMP type.

Note: This parameter can be specified only for the ICMP protocol.

Minimum value: 0

Maximum value: 65536

icmpCode

Code of a particular ICMP message type to match against the ICMP code of an incoming IPv6 ICMP packet. For example, to block DESTINATION HOST UNREACHABLE messages, specify 3 as the ICMP type and 1 as the ICMP code.

If you set this parameter, you must set the ICMP Type parameter.

Minimum value: 0

Maximum value: 65536

vlan

ID of the VLAN. The NetScaler appliance applies the ACL6 rule only to the incoming packets on the specified VLAN. If you do not specify a VLAN ID, the appliance applies the ACL6 rule to the incoming packets on all VLANs.

Minimum value: 1

Maximum value: 4094

vxlan

ID of the VXLAN. The NetScaler appliance applies the ACL6 rule only to the incoming packets on the specified VXLAN. If you do not specify a VXLAN ID, the appliance applies the ACL6 rule to the incoming packets on all VXLANs.

Minimum value: 1

Maximum value: 16777215

interface

ID of an interface. The NetScaler appliance applies the ACL6 rule only to the incoming packets from the specified interface. If you do not specify any value, the appliance applies the ACL6 rule to the incoming packets from all interfaces.

priority

Priority for the ACL6 rule, which determines the order in which it is evaluated relative to the other ACL6 rules. If you do not specify priorities while creating ACL6 rules, the ACL6 rules are evaluated in the order in which they are created.

Minimum value: 1

Maximum value: 80000

established

Allow only incoming TCP packets that have the ACK or RST bit set if the action set for the ACL6 rule is ALLOW and these packets match the other conditions in the ACL6 rule.

Example

```
set ns acl6 rule1 -srcPort 50
```

unset ns acl6

Resets the attributes of the specified ACL6 rule. To commit this operation, you must apply the ACL6s.Attributes for which a default value is available revert to their default values. Refer to the set ns acl6 command for descriptions of the parameters..Refer to the set ns acl6 command for meanings of the arguments.

Synopsis

```
unset ns acl6 <acl6name> [-srcIPv6] [-srcPort] [-destIPv6] [-destPort] [-srcMac] [-protocol] [-icmpType] [-icmpCode] [-vlan] [-vxlan] [-interface] [-established]
```

Example

```
unset ns acl6 rule1 -srcPort
```

enable ns acl6

Enables an ACL6 rule. To commit this operation, you must apply the ACL6s.After you apply the ACL6 rules, the NetScaler appliance compares incoming IPv6 packets to the enabled ACL6 rules.

Synopsis

```
enable ns acl6 <acl6name> ...
```

Arguments

acl6name

Name of ACL6 rule that you want to enable.

Example

```
enable ns acl6 rule1
```

disable ns acl6

Disables an ACL6 rule. To commit this operation, you must apply the ACL6s. After you apply the ACL6 rules, the NetScaler appliance does not compare incoming IPv6 packets to the disabled ACL6 rules.

Synopsys

```
disable ns acl6 <acl6name> ...
```

Arguments

acl6name

Name of ACL6 rule that you want to disable.

Example

```
disable ns acl6 rule1
```

stat ns acl6

Displays statistics related to the ACL6 rules. To display statistics of all the ACL6 rules, run the command without any parameters. To display statistics of a particular ACL6 rule, specify the name of the ACL6 rule.

Synopsys

```
stat ns acl6 [<acl6name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full )]
```

Arguments

acl6name

Name of the ACL6 rule whose statistics you want the NetScaler appliance to display.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Bridge ACL6 hits (ACL6Bdg)

Packets matching a bridge IPv6 ACL, which is in transparent mode and bypasses service processing.

Deny ACL6 hits (ACL6Deny)

Packets dropped because they match IPv6 ACLs with processing mode set to DENY.

Allow ACL6 hits (ACL6Allow)

Packets matching IPv6 ACLs with processing mode set to ALLOW. NetScaler processes these packets.

NAT ACL6 hits (ACL6NAT)

Packets matching a NAT ACL6, resulting in a NAT session.

ACL6 hits (ACL6Hits)

Packets matching an IPv6 ACL.

ACL6 misses (ACL6Miss)

Packets not matching any IPv6 ACL.

NAT64 ACL6 hits (ACL6NAT64)

Packets matching a NAT64 ACL6, resulting in a NAT64 translation.

ACL6 Count (ACL6Count)

Total number of ACL6 rules configured.

Hits for this ACL6 (Hits)

Number of times the acl6 was hit

Example

```
stat acl6
```

rename ns acl6

Renames an ACL6 rule. To commit this operation, you must apply the ACL6s.

Synopsys

```
rename ns acl6 <acl6name> <newName>
```

Arguments

acl6name

Name of the ACL6 rule that you want to rename.

newName

New name for the ACL6 rule. Must begin with an ASCII alphabetic or underscore `[_]` character, and must contain only ASCII alphanumeric, underscore, hash `[#]`, period `[.]`, space, colon `[:]`, at `[@]`, equals `[=]`, and hyphen `[-]` characters.

Example

```
rename acl6 rule rule-new
```

show ns acl6

Displays settings related to the ACL6 rules. To display settings of all the ACL6 rules, run the command without any parameters. To display settings of a particular ACL6 rule, specify the name of the ACL6 rule.

Synopsys

```
show ns acl6 [<acl6name>]
```

Arguments

acl6name

Name of the ACL6 rule whose details you want the NetScaler appliance to display.

Outputs

acl6action

Action to perform on the incoming IPv6 packets that match the ACL6 rule.

Available settings function as follows:

- * ALLOW - The NetScaler appliance processes the packet.
- * BRIDGE - The NetScaler appliance bridges the packet to the destination without processing it.
- * DENY - The NetScaler appliance drops the packet.

srcMac

MAC address to match against the source MAC address of an incoming IPv6 packet.

stateflag

ACL6 state flag.

protocol

Protocol number in IPv6 header or name.

protocolNumber

Protocol number in IPv6 header or name.

srcPortVal

Source port (range).

destPortVal

Destination port (range).

srcIPv6Val

Source IPv6 address (range).

destIPv6Val

Destination IPv6 address (range).

vlan

ID of the VLAN. The NetScaler appliance applies the ACL6 rule only to the incoming packets on the specified VLAN. If you do not specify a VLAN ID, the appliance applies the ACL6 rule to the incoming packets on all VLANs.

vxlan

ID of the VXLAN. The NetScaler appliance applies the ACL6 rule only to the incoming packets on the specified VXLAN. If you do not specify a VXLAN ID, the appliance applies the ACL6 rule to the incoming packets on all VXLANs.

state

State of the ACL6.

kernelstate

Commit status of the ACL6.

TTL

Time left to expire ACL6 (in seconds).

icmpType

ICMP Message type to match against the message type of an incoming IPv6 ICMP packet. For example, to block DESTINATION UNREACHABLE messages, you must specify 3 as the ICMP type.

Note: This parameter can be specified only for the ICMP protocol.

icmpCode

Code of a particular ICMP message type to match against the ICMP code of an incoming IPv6 ICMP packet. For example, to block DESTINATION HOST UNREACHABLE messages, specify 3 as the ICMP type and 1 as the ICMP code.

If you set this parameter, you must set the ICMP Type parameter.

interface

ID of an interface. The NetScaler appliance applies the ACL6 rule only to the incoming packets from the specified interface. If you do not specify any value, the appliance applies the ACL6 rule to the incoming packets from all interfaces.

hits

Number of hits of this ACL6.

established

This flag indicates that the ACL6 should be used for TCP response traffic only.

priority

Priority for the ACL6 rule, which determines the order in which it is evaluated relative to the other ACL6 rules. If you do not specify priorities while creating ACL6 rules, the ACL6 rules are evaluated in the order in which they are created.

operator

Logical operator.

time

Time when this acl is applied.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

devno**count**

Example

```
show ns acl6 rule1 1)      Name: r1      Action: DENY      srcIPv4
```

ns acls

The following operations can be performed on "ns acls":

[renumber](#) | [clear](#) | [apply](#)

renumber ns acls

Renumbers the priorities of extended ACL rules to multiples of 10. To commit this operation, you must apply the extended ACLs. Enables you to assign a new extended ACL rule a priority that is between two existing, consecutively numbered priorities. For example, if two extended ACLs, ACL1 and ACL2, have priorities 2 and 3 renumbering changes those priorities to 20 and 30. You can then add ACL3 with priority 25.

Synopsys

```
renumber ns acls
```

Example

```
renumber acls
```

clear ns acls

Removes all simple ACL rules from the NetScaler appliance. This operation does not require an explicit apply.

Synopsys

```
clear ns acls
```

Example

```
clear ns acls
```

apply ns acls

Updates the extended ACL rule's memory tree (lookup table), adding any new extended ACL rules and applying any modifications to existing ACL rules. The lookup table includes the configuration of all the extended ACL rules on the NetScaler appliance. The NetScaler appliance uses the lookup table (not the configuration file) to filter the incoming IPv4 packets.

Synopsys

```
apply ns acls
```

Example

```
apply ns acls
```

ns acls6

The following operations can be performed on "ns acls6":

[clear](#) | [apply](#) | [renumber](#)

clear ns acls6

Removes all simple ACL6 rules from the NetScaler appliance. This operation does not require an explicit apply.

Synopsys

```
clear ns acls6
```

Example

```
clear ns acls6
```

apply ns acls6

Updates the ACL6 rules' memory tree (lookup table), adding any new ACL6 rules and applying any modifications to existing ACL rules. The lookup table includes the configuration of all the ACL6 rules on the NetScaler appliance. The NetScaler appliance uses the lookup table (not the configuration file) to filter the incoming IPv4 packets.

Synopsys

```
apply ns acls6
```

Example

```
apply ns acls6
```

renumber ns acls6

Renumbers the priorities of ACL6 rules to multiples of 10. To commit this operation, you must apply the ACL6s. Enables you to assign a new ACL6 rule a priority that is between two existing, consecutively numbered priorities. For example, if two ACL6s, ACL6-1 and ACL6-2, have priorities 2 and 3 renumbering changes those priorities to 20 and 30. You can then add ACL6-3 with priority 25.

Synopsys

```
renumber ns acls6
```

Example

```
renumber acls6
```

ns appflowCollector

The following operations can be performed on "ns appflowCollector":

[add](#) | [rm](#) | [show](#)

add ns appflowCollector

Add a new AppFlow collector. NOTE: This command is deprecated. This command is deprecated in favor of 'add appflow collector'

Synopsis

Arguments

name

Name of the AppFlow collector.

IPAddress

The IPv4 address of the AppFlow collector.

port

The UDP port on which the AppFlow collector is listening.

Default value: 4739

Example

```
add ns appflowCollector collector1 -IPAddress 192.168.1.40 -port 2055
```

rm ns appflowCollector

Remove an AppFlow collector. NOTE: This command is deprecated. This command is deprecated in favor of 'rm appflow collector'

Synopsis

Arguments

name

Name of an AppFlow collector.

Example

```
rm ns appflowCollector collector1
```

show ns appflowCollector

Display details of all the AppFlow collectors configured on the system. Alternatively, to view the details of a particular AppFlow collector, specify its name. NOTE: This command is deprecated. This command is deprecated in favor of 'show appflow collector'

Synopsis

Arguments

name

Name of the AppFlow collector.

Outputs

IPAddress

The IPv4 address of the AppFlow collector.

port

The UDP port on which the AppFlow collector is listening.

devno

count

stateflag

Example

```
show ns appflowCollector collector1
```


ns appflowParam

The following operations can be performed on "ns appflowParam":

[set](#) | [unset](#) | [show](#)

set ns appflowParam

Set AppFlow parameters. NOTE: This command is deprecated.

Synopsys

Arguments

templateRefresh

IPFIX template refresh interval (in seconds).

Default value: 600

Minimum value: 60

Maximum value: 3600

udpPmtu

MTU to be used for IPFIX UDP packets.

Default value: 1472

Minimum value: 128

Maximum value: 1472

httpUrl

Enable AppFlow HTTP URL logging.

Possible values: ENABLED, DISABLED

Default value: DISABLED

httpCookie

Enable AppFlow HTTP cookie logging.

Possible values: ENABLED, DISABLED

Default value: DISABLED

httpReferer

Enable AppFlow HTTP referer logging.

Possible values: ENABLED, DISABLED

Default value: DISABLED

httpMethod

Enable AppFlow HTTP method logging.

Possible values: ENABLED, DISABLED

Default value: DISABLED

httpHost

Enable AppFlow HTTP host logging.

Possible values: ENABLED, DISABLED

Default value: DISABLED

httpUserAgent

Enable AppFlow HTTP user-agent logging.

Possible values: ENABLED, DISABLED

Default value: DISABLED

clientTrafficOnly

Control whether AppFlow records should be generated only for client-side traffic.

Possible values: YES, NO

Default value: NO

Example

```
set ns appflowParam -templateRefresh 240
```

unset ns appflowParam

Use this command to remove ns appflowParam settings. Refer to the set ns appflowParam command for meanings of the arguments. NOTE: This command is deprecated.

Synopsys

show ns appflowParam

Display AppFlow parameters. NOTE: This command is deprecated. This command is deprecated in favor of 'show appflow param'

Synopsys

Outputs

templateRefresh

IPFIX template refresh interval (in seconds).

udpPmtu

MTU to be used for IPFIX UDP packets.

httpUrl

Enable AppFlow HTTP URL logging.

httpCookie

Enable AppFlow HTTP cookie logging.

httpReferer

Enable AppFlow HTTP referer logging.

httpMethod

Enable AppFlow HTTP method logging.

httpHost

Enable AppFlow HTTP host logging.

httpUserAgent

Enable AppFlow HTTP user-agent logging.

clientTrafficOnly

Control whether AppFlow records should be generated only for client-side traffic.

ns aptlicense

The following operations can be performed on "ns aptlicense":

[show](#) | [update](#)

show ns aptlicense

Synopsys

show ns aptlicense <serialNo>

Arguments

serialNo

Hardware Serial Number/License Activation Code(LAC)

Outputs

response

Response data as text blob

id

License ID

sessionId

Session ID

bindType

Bind type

countAvailable

Count

countTotal

Count

name

License name

relevance

License relevance

datePurchased

License purchase date

dateSa

License SA date

dateExp

License expiry date

features

Example

```
show ns aptlicense <hw-no/lac>
```

update ns aptlicense

Synopsys

```
update ns aptlicense <id> <sessionId> <bindType> <countAvailable> [<licenseDir>]
```

Arguments

id

License ID

sessionId

Session ID

bindType

Bind type

countAvailable

Count

licenseDir

License Directory

Example

```
update ns aptlicense key1 sessionID# HOSTNAME 1
```

ns assignment

The following operations can be performed on "ns assignment":

[add](#) | [rm](#) | [show](#) | [rename](#)

add ns assignment

Creates an assignment of a value to a variable. The variable (the left hand side) may be a singleton variable or a map with a key expression. The value (the right hand side) is computed from a default syntax expression and may be used to set the variable or may be added to or subtracted from the current value of a ulong variable or appended to a text variable. The key expression, if present, is evaluated before the value expression. The left hand side variable value may also be cleared, in which case there is no value expression.

Synopsys

```
add ns assignment <name> -variable <expression> [-set <expression> | -add <expression> | -sub <expression> | -append <expression> | -clear] [-comment <string>]
```

Arguments

name

Name for the assignment. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) hash (#), space (), at (@), equals (=), colon (:), and underscore characters. Can be changed after the assignment is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my assignment" or ?my assignment?).

variable

Left hand side of the assignment, of the form \$variable-name (for a singleton variable) or \$variable-name[key-expression], where key-expression is a default syntax expression that evaluates to a text string and provides the key to select a map entry

set

Right hand side of the assignment. The default syntax expression is evaluated and assigned to the left hand variable.

add

Right hand side of the assignment. The default syntax expression is evaluated and added to the left hand variable.

sub

Right hand side of the assignment. The default syntax expression is evaluated and subtracted from the left hand variable.

append

Right hand side of the assignment. The default syntax expression is evaluated and appended to the left hand variable.

clear

Clear the variable value. Deallocates a text value, and for a map, the text key.

comment

Comment. Can be used to preserve information about this rewrite action.

Example

```
add ns assignment set_user_privilege -var $user_privilege_map[client.ip.src.typecast_text_
```

rm ns assignment

Removes a rewrite action.

Synopsis

```
rm ns assignment <name>
```

Arguments

name

Name for the assignment. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) hash (#), space (), at (@), equals (=), colon (:), and underscore characters. Can be changed after the assignment is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my assignment" or ?my assignment?).

Example

```
rm ns assignment set_user_privilege
```

show ns assignment

Displays configured assignments.

Synopsis

```
show ns assignment [<name>]
```

Arguments

name

Name of the assignment

Outputs

stateflag

variable

Left hand side of the assignment.

set

Right hand side of the assignment, variable set to expression value.

add

Right hand side of the assignment, expression value added to variable.

sub

Right hand side of the assignment, expression value subtracted from variable.

append

Right hand side of the assignment, expression value appended to variable.

clear

Variable cleared.

hits

The number of times the action has been taken.

undefHits

The number of times the action resulted in UNDEF.

referenceCount

The number of references to the action.

comment

Comment. Can be used to preserve information about this rewrite action.

devno**count**

Example

```
show ns assignment
```

rename ns assignment

Renames an assignment.

Synopsis

```
rename ns assignment <name>@ <newName>@
```

Arguments

name

Existing name of the assignment.

newName

New name for the assignment.

Must begin with a letter, number, or the underscore character (`_`), and must contain only letters, numbers, and the hyphen (`-`), period (`.`) hash (`#`), space (), at (`@`), equals (`=`), colon (`:`), and underscore characters. Can be changed after the rewrite policy is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my assignment" or ?my assignment?).

Example

```
rename ns assignment oldname newname
```


ns config

The following operations can be performed on "ns config":

clear | **set** | **unset** | **save** | **show** | **diff**

clear ns config

Clears the NetScaler running configurations based on different levels.

Synopsys

```
clear ns config [-force] <level>
```

Arguments

force

Configurations will be cleared without prompting for confirmation.

level

Types of configurations to be cleared.

* **basic**: Clears all configurations except the following:

- NSIP, default route (gateway), MIPs, and SNIPs
- Network settings (DG, VLAN, RHI, NTP and DNS settings)
- Cluster settings
- HA node definitions
- Feature and mode settings
- nsroot password

* **extended**: Clears the same configurations as the 'basic' option. In addition, it clears the nsroot password and feature and mode settings.

* **full**: Clears all configurations except NSIP, default route, and interface settings.

Note: When you clear the configurations through the cluster IP address, by specifying the level as 'full', the cluster is deleted and all cluster nodes become standalone appliances. The 'basic' and 'extended' levels are propagated to the cluster nodes.

Possible values: basic, extended, full

set ns config

Sets the NetScaler IP address and NetScaler VLAN. To set other NetScaler parameters, use the 'set ns param' command. Note: To change the NSIP address or the NSVLAN of an appliance that is part of a cluster, first remove the appliance from the cluster, change the NSIP or the NSVLAN, and then add the appliance back to the cluster.

Synopsys

```
set ns config [-IPAddress <ip_addr> -netmask <netmask>] [-nsvlan <positive_integer> -ifnum <interface_name> ... [-tagged ( YES | NO )]]
```

Arguments

IPAddress

IP address of the NetScaler appliance. Commonly referred to as NSIP address. This parameter is mandatory to bring up the appliance.

netmask

Netmask corresponding to the IP address. This parameter is mandatory to bring up the appliance.

nsvlan

VLAN (NSVLAN) for the subnet on which the IP address resides.

Minimum value: 2

Maximum value: 4094

ifnum

Interfaces of the appliances that must be bound to the NSVLAN.

Minimum value: 1

tagged

Specifies that the interfaces will be added as 802.1q tagged interfaces. Packets sent on these interface on this VLAN will have an additional 4-byte 802.1q tag which identifies the VLAN.

To use 802.1q tagging, the switch connected to the appliance's interfaces must also be configured for tagging.

Possible values: YES, NO

Default value: YES

unset ns config

Removes the attributes of the NetScaler appliance. Attributes for which a default value is available revert to their default values. Refer to the 'set ns config' command for a description of the parameters..Refer to the set ns config command for meanings of the arguments.

Synopsys

```
unset ns config [-nsvlan] [-IPAddress] [-netmask] [-ifnum] [-tagged]
```

save ns config

Save the configurations to the appliances FLASH memory in the /nsconfig/ns.conf file. Backup configuration files are named ns.conf.n. The most recent backup file has the smallest value for n.

Synopsys

```
save ns config
```

Outputs

message

show ns config

Displays the following details of the NetScaler appliance: * NetScaler IP address and subnet mask * Number of mapped IP addresses * Identifies the appliance as a standalone appliance, a part of a HA pair, or is a cluster node * Current time on the system and timestamp when the appliance was last updated Note: To view the complete configurations that have been executed on the appliance, run the 'show ns runningConfig' command.

Synopsys

```
show ns config
```

Outputs

IPAddress

IP Address of the System.

netmask

The netmask corresponding to the IP address.

mappedIP

Mapped IP Address of the System.

range

The range of Mapped IP addresses to be configured.

nsvlan

The VLAN (NSVLAN) for the subnet on which the system IP resides.

ifnum

Bind the given ports to the NSVLAN.

tagged

Specifies that the interfaces will be added as 802.1q tagged interfaces. Packets sent on these interface on this VLAN will have an additional 4-byte 802.1q tag which identifies the VLAN.

To use 802.1q tagging, the switch connected to the appliance's interfaces must also be configured for tagging.

httpPort

The HTTP ports on the Web server.

maxConn

Maximum Number of Connections.

maxReq

Maximum Number of requests that can be handled.

cip

Insertion of client IP address into the HTTP header.

cipHeader

The text that will be used as the client IP header.

cookieversion

The version of the cookie inserted by system.

secureCookie

enable/disable secure flag for persistence cookie

failover

Standalone node.

systemType

The type of the System. Possible Values: Standalone, HA, Cluster

primaryIP

HA Master Node IP address.

pmtuMin

The minimum Path MTU.

pmtuTimeout

The timeout value in minutes.

ftpPortRange

Port range configured for FTP services.

crPortRange

Port range for cache redirection services.

flags

The flags for this entry.

timezone

Name of the timezone

LastConfigChangedTime

Time when the configuration was last modified.

LastConfigSaveTime

Time when the configuration was last saved through savensconfig.

currentSytemTime

current system time in date format.

systemTime

current system time.

grantQuotaMaxClient

The percentage of shared quota to be granted at a time for maxClient

exclusiveQuotaMaxClient

The percentage of maxClient to be given to PEs

grantQuotaSpillOver

The percentage of shared quota to be granted at a time for spillover

exclusiveQuotaSpillOver

The percentage of max limit to be given to PEs

nfwfmode

Network Firewallmode

diff ns config

Difference between two configuration

Synopsys

```
diff ns config [<config1>] [<config2>] [-outtype ( cli | xml )] [-template] [-ignoreDeviceSpecific]
```

Arguments

config1

Location of the configurations.

config2

Location of the configurations.

outtype

Format to display the difference in configurations.

Possible values: cli, xml

template

File that contains the commands to be compared.

ignoreDeviceSpecific

Suppress device specific differences.

Outputs

response

Example

Generates the differences between two configurations. Note: If no parameters are provided

ns connectiontable

The following operations can be performed on "ns connectiontable":

show ns connectiontable

Displays the current TCP/IP connection table.

Synopsys

show ns connectiontable [<filterexpression>] [-detail <detail> ...]

Arguments

filterexpression

The maximum length of filter expression is 255 and it can be of following format:

<expression> [<relop> <expression>]

<relop> = (&& | ||)

connectiontable supports two types of filter expressions:

Classic Expressions:

<expression> = the expression string in the format:

<qualifier> <operator> <qualifier-value>

<qualifier> = SOURCEIP.

<qualifier-value> = A valid IP address.

<qualifier> = SOURCEPORT.

<qualifier-value> = A valid port number.

<qualifier> = DESTIP.

<qualifier-value> = A valid IP address.

<qualifier> = DESTPORT.

<qualifier-value> = A valid port number.

<qualifier> = IP.

<qualifier-value> = A valid IP address.

<qualifier> = PORT.

<qualifier-value> = A valid port number.

<qualifier> = IDLETIME.

<qualifier-value> = A positive integer indicating the idle time.

<qualifier> = SVCNAME.

<qualifier-value> = The name of a service.

<qualifier> = VSVRNAME.

<qualifier-value> = The name of a vserver.

<qualifier> = CONNID

<qualifier-value> = A valid PCB dev number.

<qualifier> = INTF

<qualifier-value> = A valid interface id in the form of x/y

(n/x/y in case of cluster interface).

<qualifier> = VLAN

<qualifier-value> = A valid VLAN ID.

<qualifier> = STATE.

<qualifier-value> = (CLOSE_WAIT | CLOSED | CLOSING | ESTABLISHED |

FIN_WAIT_1 | FIN_WAIT_2 | LAST_ACK | LISTEN |

SYN_RECEIVED | SYN_SENT | TIME_WAIT)

<qualifier> = SVCTYPE.

<qualifier-value> = (HTTP | FTP | TCP | UDP | SSL |

SSL_BRIDGE | SSL_TCP | NNTP | RPCSVR | RPCSVRS |

RPCCLNT | DNS | ADNS | SNMP | RTSP | DHCPRA | ANY |

MONITOR | MONITOR_UDP | MONITOR_PING | SIP_UDP | MYSQL | MSSQL | UNKNOWN)

<operator> = (== | eq | != | neq | > | gt | < | lt | >= |

ge | <= | le | BETWEEN)

Default Expressions:

<expression> =:

CONNECTION.<qualifier>.<qualifier-method>.<qualifier-value>

<qualifier> = SRCIP

<qualifier-method> = [EQ | NE]

<qualifier-value> = A valid IPv4 address

example = CONNECTION.SRCIP.EQ(127.0.0.1)

<qualifier> = DSTIP

<qualifier-method> = [EQ | NE]

<qualifier-value> = A valid IPv4 address.

example = CONNECTION.DSTIP.EQ(127.0.0.1)

<qualifier> = IP

<qualifier-method> = [EQ | NE]

<qualifier-value> = A valid IPv4 address.

example = CONNECTION.IP.EQ(127.0.0.1)

<qualifier> = SRCIPv6

<qualifier-method> = [EQ | NE]

<qualifier-value> = A valid IPv6 address.

example = CONNECTION.SRCIPv6.EQ(2001:db8:0:0:1::1)

<qualifier> = DSTIPv6

<qualifier-method> = [EQ | NE]

<qualifier-value> = A valid IPv6 address.

example = CONNECTION.DSTIPv6.EQ(2001:db8:0:0:1::1)

<qualifier> = IPv6

<qualifier-method> = [EQ | NE]

<qualifier-value> = A valid IPv6 address.

example = CONNECTION.IPv6.EQ(2001:db8:0:0:1::1)

<qualifier> = SRCPORT

<qualifier-method> = [EQ | NE | GT | GE | LT | LE

| BETWEEN]

<qualifier-value> = A valid port number.

example = CONNECTION.SRCPORT.EQ(80)

<qualifier> = DSTPORT

<qualifier-method> = [EQ | NE | GT | GE | LT | LE

| BETWEEN]

<qualifier-value> = A valid port number.

example = CONNECTION.DSTPORT.EQ(80)

<qualifier> = PORT

<qualifier-method> = [EQ | NE | GT | GE | LT | LE

| BETWEEN]

<qualifier-value> = A valid port number.

example = CONNECTION.PORT.EQ(80)

<qualifier> = SVCNAME

<qualifier-method> = [EQ | NE | CONTAINS | STARTSWITH

| ENDSWITH]

<qualifier-value> = service name.

example = CONNECTION.SVCNAME.EQ("name")

<qualifier> = LB_VSERVER.NAME

<qualifier-method> = [EQ | NE | CONTAINS | STARTSWITH

| ENDSWITH]

<qualifier-value> = LB vserver name.

example = CONNECTION.LB_VSERVER.NAME.EQ("name")

<qualifier> = CS_VSERVER.NAME

<qualifier-method> = [EQ | NE | CONTAINS | STARTSWITH

| ENDSWITH]

<qualifier-value> = CS vserver name.

example = CONNECTION.CS_VSERVER.NAME.EQ("name")

<qualifier> = INTF

<qualifier-method> = [EQ | NE]

<qualifier-value> = A valid interface id in the form of x/y (n/x/y in case of cluster interface).

example = CONNECTION.INTF.EQ("0/1/1")

<qualifier> = VLANID

<qualifier-method> = [EQ | NE | GT | GE | LT | LE | BETWEEN]

<qualifier-value> = A valid VLAN ID.

example = CONNECTION.VLANID.EQ(0)

<qualifier> = CONNID

<qualifier-method> = [EQ | NE | GT | GE | LT | LE | BETWEEN]

<qualifier-value> = A valid PCB dev number.

example = CONNECTION.CONNID.EQ(0)

<qualifier> = PPEID

<qualifier-method> = [EQ | NE | GT | GE | LT | LE | BETWEEN]

<qualifier-value> = A valid core ID.

example = CONNECTION.PPEID.EQ(0)

<qualifier> = IDLETIME

<qualifier-method> = [EQ | NE | GT | GE | LT | LE | BETWEEN]

<qualifier-value> = A positive integer indicating the idletime.

example = CONNECTION.IDLETIME.LT(100)

<qualifier> = TCPSTATE

<qualifier-method> = [EQ | NE]

<qualifier-value> = (CLOSE_WAIT | CLOSED | CLOSING | ESTABLISHED | FIN_WAIT_1 | FIN_WAIT_2 | LAST_ACK | LISTEN | SYN_RECEIVED | SYN_SENT | TIME_WAIT | NOT_APPLICABLE)

example = CONNECTION.TCPSTATE.EQ(LISTEN)

<qualifier> = SERVICE_TYPE

<qualifier-method> = [EQ | NE]

<qualifier-value> = (SVC_HTTP | FTP | TCP | UDP | SSL |
SSL_BRIDGE | SSL_TCP | NNTP | RPCSVR | RPCSVRS |
RPCCLNT | SVC_DNS | ADNS | SNMP | RTSP | DHCPRA | ANY |
MONITOR | MONITOR_UDP | MONITOR_PING | SIP_UDP |
SVC_MYSQL | SVC_MSSQL | SERVICE_UNKNOWN)

example = CONNECTION.SERVICE_TYPE.EQ(ANY)

<qualifier> = TRAFFIC_DOMAIN_ID

<qualifier-method> = [EQ | NE | GT | GE | LT | LE
| BETWEEN]

<qualifier-value> = A valid traffic domain ID.

example = CONNECTION.TRAFFIC_DOMAIN_ID.EQ(0)

common usecases:

Filtering out loopback connections and view present

connections through netslaer

show connectiontable "CONNECTION.IP.NEQ(127.0.0.1) &&

CONNECTION.TCPSTATE.EQ(ESTABLISHED)" -detail full

show connections from a particular sourceip and targeted

to port 80

show connectiontable "CONNECTION.SRCIP.EQ(10.102.1.91) &&

CONNECTION.DSTPORT.EQ(80)"

show connection particular to a service and its linked

client connections

show connectiontable "CONNECTION.SVCNAME.EQ("S1")"

-detail link

show connections for a particular servicetype(e.g.http)

show connectiontable "CONNECTION.SERVICE_TYPE.EQ(TCP)"

viewing connections that have been idle for a long time

show connectiontable "CONNECTION.IDLETIME.GT(100)"

show connections for a particular interface and vlan

show connectiontable "CONNECTION.INTF.EQ("1/1") &&

CONNECTION.VLANID.EQ(1)"

detail

Specify display options for the connection table.

* LINK - Displays the linked PCB (Protocol Control Block).

* NAME - Displays along with the service name.

* CONNFAILOVER - Displays PCB with connection failover.

* FULL - Displays all available details.

Outputs

SOURCEIP

Source IP of the connection.

SOURCEPORT

Source port of the connection.

DESTIP

Destination IP of the connection.

DESTPORT

Destination port of the connection.

SVCTYPE

Protocol supported by the connection.

IDLETIME

Time since last activity was detected on the connection.

STATE

Current TCP/IP state of the connection.

linkSourceIP

Source IP of the link connection.

linkSourcePort

Source port of the link connection.

linkDestIP

Destination IP of the link connection.

linkDestPort

Destination port of the link connection.

linkServiceType

Protocol supported by the link connection.

linkIdleTime

Time since last activity was detected on link connection.

linkState

TCP/IP current state of link connection.

entityName

NetScaler entity name for the connection.

linkEntityName

NetScaler entity name for link connection.

connectionNumber

Connection number

linkConnectionNumber

Link connection number

connid

Unique transaction number for the connection.

linkConnid

Unique transaction number for the peer connection.

filterFlags

flags used to store display options

optionFlags

flags used to store TCP options like Sack, WS

nsWSvalue

netscaler window scaling value

peerWSvalue

peer window scaling value

mss

Client side MSS for the connection - used in server SYN.

retxRetryCnt

Retransmission retry count for the connection.

rcvWnd

Received Advertised Window for the connection.

advWnd

Sent advertised window for the connection.

sndCwnd

sent congestion window for the connection.

iss

Initial send sequence number for the connection.

irs

Initial receive sequence number for the connection.

rcvNxt

next expecting seq number for the connection.

maxAck

current running max ack sent for the connection.

sndNxt

next bytes seq number for the connection.

sndUnAck

Most recently received ACK for the connection.

httpEndSeq

HTTP parsing tracking seq number for the connection.

httpState

HTTP Protocol state for the connection.

trCount

Max reuests allowed per connection.

priority

priority of the connection.

httpReqVer

current HTTP request version on the connection.

httpRequest

current HTTP request type on the connection.

httpRspCode

current response type on the connection.

rttSmoothed

smoothed RTT value of the connection.

rttVariance

RTT variance for the connection.

outoforderPkts

held packets on the connection.

count**linkOptionFlag**

Link connection's TCP option flag for Sack and WS

linknsWSvalue

Link connection-s netscaler window scaling value

linkpeerWSvalue

Link connection-s peer netscaler window scaling value

linkMSS

Client side MSS for the Link connection - used in server SYN

linkRetxRetryCnt

Retransmission retry count for the Link connection.

linkRcvWnd

Received Advertised Window for the Link connection.

linkAdvWnd

Sent advertised window for the Link connection.

linkSndCwnd

Send congestion window for the Link connection.

linkISS

Initial send seq number for the Link connection.

linkIRS

Initial receive seq number for the Link connection.

linkRcvNxt

Next expecting seq number on the Link connection.

linkMaxAck

Current running maximum ack sent on the Link connection.

linkSndNxt

Next bytes seq number for the Link connection.

linkSndUnAck

Most recently received ACK on the Link connection.

linkHttpEndSeq

HTTP parsing tracking seq number on the Link connection.

linkHttpState

HTTP protocol state on the Link connection.

linkTrCount

Max requests per connection for Link connection.

linkPriority

Priority for the Link connection.

linkHttpRequestVer

HTTP current request version on Link connection.

linkHttpRequest

HTTP current request type on Link connection.

linkHttpRspCode

Current response type on link connection.

linkRttSmoothed

Smoothed RTT value on link connection.

linkRttVariance

RTT variance on Link connection.

linkHeldPkts

Held packets on Link connection.

targetnodeidnm

NNM connection target node ID.

sourcenodeidnnm

NNM connection source node ID.

channelidnnm

NNM connection channel ID.

msgversionnnm

nnm message version.

td

Traffic Domain Id.

devno

stateflag

ns consoleloginprompt

The following operations can be performed on "ns consoleloginprompt":

[set](#) | [unset](#) | [show](#)

set ns consoleloginprompt

Synopsys

```
set ns consoleloginprompt <promptString>
```

Arguments

promptString

Console login prompt string

Example

```
set ns consoleloginprompt <prompt_string>
```

unset ns consoleloginprompt

Use this command to remove ns consoleloginprompt settings. Refer to the set ns consoleloginprompt command for meanings of the arguments.

Synopsys

```
unset ns consoleloginprompt -promptString
```

show ns consoleloginprompt

Synopsys

```
show ns consoleloginprompt
```

Outputs

promptString

Console login prompt string

Example

```
get ns consoleloginprompt
```


ns dhcplp

The following operations can be performed on "ns dhcplp":

release ns dhcplp

Releases the IP address acquired by the DHCP client.

Synopsys

```
release ns dhcplp
```

ns dhcpParams

The following operations can be performed on "ns dhcpParams":

[set](#) | [unset](#) | [show](#)

set ns dhcpParams

Sets the DHCP client parameters.

Synopsys

```
set ns dhcpParams [-dhcpClient ( ON | OFF )] [-saveroute ( ON | OFF )]
```

Arguments

dhcpClient

Enables DHCP client to acquire IP address from the DHCP server in the next boot. When set to OFF, disables the DHCP client in the next boot.

Possible values: ON, OFF

Default value: OFF

saveroute

DHCP acquired routes are saved on the NetScaler appliance.

Possible values: ON, OFF

Default value: OFF

unset ns dhcpParams

Use this command to remove ns dhcpParams settings. Refer to the set ns dhcpParams command for meanings of the arguments.

Synopsys

```
unset ns dhcpParams [-dhcpClient] [-saveroute]
```

show ns dhcpParams

Displays the parameters configured for the DHCP client.

Synopsys

```
show ns dhcpParams
```

Outputs

dhcpClient

ON, if DHCP client active on next reboot, else OFF

IPAddress

DHCP acquired IP

netmask

DHCP acquired Netmask

hostRtGw

DHCP acquired Gateway

running

DHCP mode

saveroute

DHCP acquired gateway save flag

ns diameter

The following operations can be performed on "ns diameter":

[set](#) | [unset](#) | [show](#)

set ns diameter

Set the diameter configuration on NS.

Synopsys

```
set ns diameter [-identity <string>] [-realm <string>] [-serverClosePropagation ( YES | NO )]
```

Arguments

identity

DiameterIdentity to be used by NS. DiameterIdentity is used to identify a Diameter node uniquely. Before setting up diameter configuration, Netscaler (as a Diameter node) MUST be assigned a unique DiameterIdentity.

example =>

```
set ns diameter -identity netscaler.com
```

Now whenever Netscaler system needs to use identity in diameter messages. It will use 'netscaler.com' as Origin-Host AVP as defined in RFC3588

realm

Diameter Realm to be used by NS.

example =>

```
set ns diameter -realm com
```

Now whenever Netscaler system needs to use realm in diameter messages. It will use 'com' as Origin-Realm AVP as defined in RFC3588

serverClosePropagation

when a Server connection goes down, whether to close the corresponding client connection if there were requests pending on the server.

Possible values: YES, NO

Default value: NO

unset ns diameter

Use this command to remove ns diameter settings. Refer to the set ns diameter command for meanings of the arguments.

Synopsys

```
unset ns diameter -serverClosePropagation
```

show ns diameter

Displays the diameter parameters configured on the NetScaler appliance.

Synopsys

```
show ns diameter
```

Outputs

identity

DiameterIdentity to be used by NS. DiameterIdentity is used to identify a Diameter node uniquely. Before setting up diameter configuration, Netscaler (as a Diameter node) MUST be assigned a unique DiameterIdentity.

example =>

```
set ns diameter -identity netscaler.com
```

Now whenever Netscaler system needs to use identity in diameter messages. It will use 'netscaler.com' as Origin-Host AVP as defined in RFC3588

realm

Diameter Realm to be used by NS.

example =>

```
set ns diameter -realm com
```

Now whenever Netscaler system needs to use realm in diameter messages. It will use 'com' as Origin-Realm AVP as defined in RFC3588

serverClosePropagation

when a Server connection goes down, whether to close the corresponding client connection if there were requests pending on the server.

ns encryptionParams

The following operations can be performed on "ns encryptionParams":

[set](#) | [show](#)

set ns encryptionParams

Sets the parameters required for encrypting or decrypting content.

Synopsis

```
set ns encryptionParams -method <method> [-keyValue ]
```

Arguments

method

Cipher method (and key length) to be used to encrypt and decrypt content. The default value is AES256.

Possible values: NONE, RC4, DES3, AES128, AES192, AES256

keyValue

The base64-encoded key generation number, method, and key value.

Note:

- * Do not include this argument if you are changing the encryption method.

- * To generate a new key value for the current encryption method, specify an empty string `\\(""\\)` as the value of this parameter. The parameter is passed implicitly, with its automatically generated value, to the NetScaler packet engines even when it is not included in the command. Passing the parameter to the packet engines enables the appliance to save the key value to the configuration file and to propagate the key value to the secondary appliance in a high availability setup.

Example

```
set ns encryptionParams -method aes128
```

show ns encryptionParams

Displays the encryption method configured on the NetScaler appliance.

Synopsis

```
show ns encryptionParams
```

Outputs

method

The cipher method (and key length) used to encrypt and decrypt content.

keyValue

The base64-encoded key generation number, method, and key value.

Note:

- * Do not include this argument if you are changing the encryption method.

- * To generate a new key value for the current encryption method, specify an empty string `\\(""\\)` as the value of this parameter. The parameter is passed implicitly, with its automatically generated value, to the NetScaler packet engines even when it is not included in the command. Passing the parameter to the packet engines

enables the appliance to save the key value to the configuration file and to propagate the key value to the secondary appliance in a high availability setup.

ns events

The following operations can be performed on "ns events":

show ns events

Displays events that occur on the appliance.

Synopsys

show ns events [<eventNo>]

Arguments

eventNo

Event number starting from which events must be shown.

Minimum value: 0

Outputs

time

Event no.

eventcode

event Code.

devid

Device Name.

devname

Device Name.

text

Event no.

data0

additional event information.

data1

additional event information.

data2

additional event information.

data3

additional event information.

devno

count

stateflag

Example

show ns events

ns feature

The following operations can be performed on "ns feature":

[enable](#) | [disable](#) | [show](#)

enable ns feature

Enables NetScaler feature(s).

Synopsys

```
enable ns feature <feature> ...
```

Arguments

feature

Feature to be enabled. Multiple features can be specified by providing a blank space between each feature.

Example

```
enable ns feature sc This CLI command enables the SureConnect feature.
```

disable ns feature

Disables NetScaler feature(s).

Synopsys

```
disable ns feature <feature> ...
```

Arguments

feature

Feature to be disabled. Multiple features can be specified by providing a blank space between each feature.

show ns feature

Displays the current state of NetScaler features.

Synopsys

```
show ns feature
```

Outputs

feature

Feature to be enabled. Multiple features can be specified by providing a blank space between each feature.

WL

Web Logging.

SP

Surge Protection.

LB

Load Balancing.

CS

Content Switching.

CR

Cache Redirect.

SC

Sure Connect.

CMP

Compression.

PQ

Priority Queuing.

SSL

Secure Sockets Layer.

GSLB

Global Server Load Balancing.

HDOSP

DOS Protection.

Routing

Routing.

CF

Content Filter.

IC

Integrated Caching.

SSLVPN

SSL VPN.

AAA

AAA

OSPF

OSPF Routing.

RIP

RIP Routing.

BGP

BGP Routing.

REWRITE

Rewrite.

IPv6PT

IPv6 protocol translation

AppFw

Application Firewall.

RESPONDER

Responder.

HTMLInjection

HTML Injection.

push

NetScaler Push.

AppFlow

AppFlow.

CloudBridge

CloudBridge.

ISIS

ISIS Routing.

CH

Call Home.

AppQoS

AppQoS

DiskCaching

Integrated Disk Cache

vPath

Vpath

ContentAccelerator

Transparent Integrated Caching.

RISE

RISE

FEO

Optimize Web content (HTML, CSS, JavaScript, images)

ns hardware

The following operations can be performed on "ns hardware":

show ns hardware

Displays details of the appliance hardware and information such as the host ID and the serial number.

Synopsys

show ns hardware

Outputs

hwdescription

Hardware and it's ports detail.

sysId

System id.

manufactureDay

Manufacturing day.

manufactureMonth

Manufacturing month.

manufactureYear

Manufacturing year.

cpufrequency

CPU Frequency.

hostId

host id.

host

host id.

serialNo

Serial no.

encodedSerialNo

Encoded serial no.

ns hostName

The following operations can be performed on "ns hostName":

[set](#) | [show](#)

set ns hostName

Sets the hostname for the NetScaler appliance. The hostname is displayed on the shell prompt.

Synopsys

```
set ns hostName <hostName> [-ownerNode <positive_integer>]
```

Arguments

hostName

Host name for the NetScaler appliance.

ownerNode

ID of the cluster node for which you are setting the hostname. Can be configured only through the cluster IP address.

Default value: 255

Minimum value: 0

Maximum value: 31

Example

```
set ns hostname nspr1
```

show ns hostName

Displays the host name of the system.

Synopsys

```
show ns hostName
```

Outputs

hostName

Host name

ownerNode

ID of the cluster node for which you are setting the hostname. Can be configured only through the cluster IP address.

devno

count

stateflag

Example

```
show ns hostname
```

ns httpParam

The following operations can be performed on "ns httpParam":

[set](#) | [unset](#) | [show](#)

set ns httpParam

Sets the configurable HTTP parameters for the NetScaler appliance.

Synopsys

```
set ns httpParam [-dropInvalReqs ( ON | OFF )] [-markHttp09Inval ( ON | OFF )] [-markConnReqInval ( ON | OFF )] [-insNsSrvrHdr ( ON | OFF ) [<nsSrvrHdr>]] [-logErrResp ( ON | OFF )] [-conMultiplex ( ENABLED | DISABLED )] [-maxReusePool <positive_integer>]
```

Arguments

dropInvalReqs

Drop invalid HTTP requests or responses.

Possible values: ON, OFF

Default value: OFF

markHttp09Inval

Mark HTTP/0.9 requests as invalid.

Possible values: ON, OFF

Default value: OFF

markConnReqInval

Mark CONNECT requests as invalid.

Possible values: ON, OFF

Default value: OFF

insNsSrvrHdr

Enable or disable NetScaler server header insertion for NetScaler generated HTTP responses.

Possible values: ON, OFF

Default value: OFF

nsSrvrHdr

The server header value to be inserted. If no explicit header is specified then NSBUILD.RELEASE is used as default server header.

logErrResp

Server header value to be inserted.

Possible values: ON, OFF

Default value: ON

conMultiplex

Reuse server connections for requests from more than one client connections.

Possible values: ENABLED, DISABLED

Default value: ENABLED

maxReusePool

Maximum limit on the number of connections, from the NetScaler to a particular server that are kept in the reuse pool. This setting is helpful for optimal memory utilization and for reducing the idle connections to the server just after the peak time.

Minimum value: 0

Maximum value: 360000

Example

```
set ns httpParam -dropInvalReqs ON
```

unset ns httpParam

Use this command to remove ns httpParam settings. Refer to the set ns httpParam command for meanings of the arguments.

Synopsys

```
unset ns httpParam [-dropInvalReqs] [-markHttp09Inval] [-markConnReqInval] [-insNsSvrHdr] [-nsSvrHdr] [-logErrResp] [-conMultiplex] [-maxReusePool]
```

show ns httpParam

Displays the HTTP parameters configured on the NetScaler appliance.

Synopsys

```
show ns httpParam
```

Outputs

dropInvalReqs

Drop invalid HTTP requests or responses.

markHttp09Inval

Mark HTTP/0.9 requests as invalid.

markConnReqInval

Mark CONNECT requests as invalid.

insNsSvrHdr

Enable or disable NetScaler server header insertion for NetScaler generated HTTP responses.

nsSvrHdr

The server header value to be inserted. If no explicit header is specified then NSBUILD.RELEASE is used as default server header.

logErrResp

Whether to log HTTP error responses

conMultiplex

Reuse server connections for requests from more than one client connections.

maxReusePool

Maximum limit on the number of connections, from the NetScaler to a particular server that are kept in the reuse pool. This setting is helpful for optimal memory utilization and for reducing the idle connections to the server just after the peak time.

ns httpProfile

The following operations can be performed on "ns httpProfile":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add ns httpProfile

Adds an HTTP profile to the NetScaler appliance.

Synopsys

```
add ns httpProfile <name> [-dropInvalReqs ( ENABLED | DISABLED )] [-markHttp09Inval ( ENABLED | DISABLED )] [-markConnReqInval ( ENABLED | DISABLED )] [-cmpOnPush ( ENABLED | DISABLED )] [-conMultiplex ( ENABLED | DISABLED )] [-maxReusePool <positive_integer>] [-dropExtraCRLF ( ENABLED | DISABLED )] [-incompHdrDelay <positive_integer>] [-webSocket ( ENABLED | DISABLED )] [-rtspTunnel ( ENABLED | DISABLED )] [-reqTimeout <positive_integer>] [-adptTimeout ( ENABLED | DISABLED )] [-reqTimeoutAction <string>] [-dropExtraData ( ENABLED | DISABLED )] [-webLog ( ENABLED | DISABLED )] [-clientIpHdrExpr <expression>] [-maxReq <positive_integer>] [-persistentETag ( ENABLED | DISABLED )] [-spdy <spdy>] [-reusePoolTimeout <positive_integer>] [-maxHeaderLen <positive_integer>]
```

Arguments

name

Name for an HTTP profile. Must begin with a letter, number, or the underscore `\\(_\\)` character. Other characters allowed, after the first character, are the hyphen `\\(-\\)`, period `\\(.\\)`, hash `\\(\\#\\)`, space `\\(\\)`, at `\\(@\\)`, and equal `\\(=\\)` characters. The name of a HTTP profile cannot be changed after it is created.

CLI Users: If the name includes one or more spaces, enclose the name in double or single quotation marks `\\` (for example, "my http profile" or 'my http profile').

dropInvalReqs

Drop invalid HTTP requests or responses.

Possible values: ENABLED, DISABLED

Default value: DISABLED

markHttp09Inval

Mark HTTP/0.9 requests as invalid.

Possible values: ENABLED, DISABLED

Default value: DISABLED

markConnReqInval

Mark CONNECT requests as invalid.

Possible values: ENABLED, DISABLED

Default value: DISABLED

cmpOnPush

Start data compression on receiving a TCP packet with PUSH flag set.

Possible values: ENABLED, DISABLED

Default value: DISABLED

conMultiplex

Reuse server connections for requests from more than one client connections.

Possible values: ENABLED, DISABLED

Default value: ENABLED

maxReusePool

Maximum limit on the number of connections, from the NetScaler to a particular server that are kept in the reuse pool. This setting is helpful for optimal memory utilization and for reducing the idle connections to the server just after the peak time.

Minimum value: 0

Maximum value: 360000

dropExtraCRLF

Drop any extra 'CR' and 'LF' characters present after the header.

Possible values: ENABLED, DISABLED

Default value: ENABLED

incompHdrDelay

Maximum time to wait, in milliseconds, between incomplete header packets. If the header packets take longer to arrive at NetScaler, the connection is silently dropped.

Default value: 7000

Minimum value: 0

Maximum value: 360000

webSocket

HTTP connection to be upgraded to a web socket connection. Once upgraded, NetScaler does not process Layer 7 traffic on this connection.

Possible values: ENABLED, DISABLED

Default value: DISABLED

rtspTunnel

Allow RTSP tunnel in HTTP. Once application/x-rtsp-tunnelled is seen in Accept or Content-Type header, NetScaler does not process Layer 7 traffic on this connection.

Possible values: ENABLED, DISABLED

Default value: DISABLED

reqTimeout

Time, in seconds, within which the HTTP request must complete. If the request does not complete within this time, the specified request timeout action is executed.

Minimum value: 0

Maximum value: 86400

adptTimeout

Adapts the configured request timeout based on flow conditions. The timeout is increased or decreased internally and applied on the flow.

Possible values: ENABLED, DISABLED

Default value: DISABLED

reqTimeoutAction

Action to take when the HTTP request does not complete within the specified request timeout duration. You can configure the following actions:

- * RESET - Send RST (reset) to client when timeout occurs.

- * DROP - Drop silently when timeout occurs.

- * Custom responder action - Name of the responder action to trigger when timeout occurs, used to send custom message.

dropExtraData

Drop any extra data when server sends more data than the specified content-length.

Possible values: ENABLED, DISABLED

Default value: DISABLED

webLog

Enable or disable web logging.

Possible values: ENABLED, DISABLED

Default value: ENABLED

clientIpHdrExpr

Name of the header that contains the real client IP address.

maxReq

Maximum requests allowed on a single connection.

Default value: 0

Minimum value: 0

Maximum value: 65534

persistentETag

Generate the persistent NetScaler specific ETag for the HTTP response with ETag header.

Possible values: ENABLED, DISABLED

Default value: DISABLED

spdy

Enable SPDYv2 or SPDYv3 or both over SSL vserver. SSL will advertise SPDY support during NPN Handshake. Both SPDY versions are enabled when this parameter is set to BOTH.

Possible values: DISABLED, ENABLED, V2, V3

Default value: DISABLED

reusePoolTimeout

Idle timeout (in seconds) for server connections in re-use pool. Connections in the re-use pool are flushed, if they remain idle for the configured timeout.

Default value: 0

Minimum value: 0

Maximum value: 31536000

maxHeaderLen

Number of bytes to be queued to look for complete header before returning error. If complete header is not obtained after queuing these many bytes, request will be marked as invalid and no L7 processing will be done for that TCP connection.

Default value: 24820

Minimum value: 2048

Maximum value: 61440

Example

```
add httpprofile <profile name> -dropInvalReqs ON -markHttp09Inval ON
```

rm ns httpProfile

Removes an HTTP profile from the appliance.

Synopsys

```
rm ns httpProfile <name>
```

Arguments

name

Name of the HTTP profile to be removed.

Example

```
rm httpprofile <profile name>
```

set ns httpProfile

Modifies the attributes of an HTTP profile.

Synopsys

```
set ns httpProfile <name> [-dropInvalReqs ( ENABLED | DISABLED )] [-markHttp09Inval ( ENABLED | DISABLED )] [-markConnReqInval ( ENABLED | DISABLED )] [-cmpOnPush ( ENABLED | DISABLED )] [-conMultiplex ( ENABLED | DISABLED )] [-maxReusePool <positive_integer>] [-dropExtraCRLF ( ENABLED | DISABLED )] [-incompHdrDelay <positive_integer>] [-webSocket ( ENABLED | DISABLED )] [-rtspTunnel ( ENABLED | DISABLED )] [-reqTimeout <positive_integer>] [-adptTimeout ( ENABLED | DISABLED )] [-reqTimeoutAction <string>] [-dropExtraData ( ENABLED | DISABLED )] [-webLog ( ENABLED | DISABLED )] [-clientIpHdrExpr <expression>] [-maxReq <positive_integer>] [-persistentETag ( ENABLED | DISABLED )] [-spdy <spdy>] [-reusePoolTimeout <positive_integer>] [-maxHeaderLen <positive_integer>]
```

Arguments

name

Name of the HTTP profile to be modified.

dropInvalReqs

Drop invalid HTTP requests or responses.

Possible values: ENABLED, DISABLED

Default value: DISABLED

markHttp09Inval

Mark HTTP/0.9 requests as invalid.

Possible values: ENABLED, DISABLED

Default value: DISABLED

markConnReqInval

Mark CONNECT requests as invalid.

Possible values: ENABLED, DISABLED

Default value: DISABLED

cmpOnPush

Start data compression on receiving a TCP packet with PUSH flag set.

Possible values: ENABLED, DISABLED

Default value: DISABLED

conMultiplex

Reuse server connections for requests from more than one client connections.

Possible values: ENABLED, DISABLED

Default value: ENABLED

maxReusePool

Maximum limit on the number of connections, from the NetScaler to a particular server that are kept in the reuse pool. This setting is helpful for optimal memory utilization and for reducing the idle connections to the server just after the peak time.

Minimum value: 0

Maximum value: 360000

dropExtraCRLF

Drop any extra 'CR' and 'LF' characters present after the header.

Possible values: ENABLED, DISABLED

Default value: ENABLED

incompHdrDelay

Maximum time to wait, in milliseconds, between incomplete header packets. If the header packets take longer to arrive at NetScaler, the connection is silently dropped.

Default value: 7000

Minimum value: 0

Maximum value: 360000

webSocket

HTTP connection to be upgraded to a web socket connection. Once upgraded, NetScaler does not process Layer 7 traffic on this connection.

Possible values: ENABLED, DISABLED

Default value: DISABLED

rtspTunnel

Allow RTSP tunnel in HTTP. Once application/x-rtsp-tunnelled is seen in Accept or Content-Type header, NetScaler does not process Layer 7 traffic on this connection.

Possible values: ENABLED, DISABLED

Default value: DISABLED

reqTimeout

Time, in seconds, within which the HTTP request must complete. If the request does not complete within this time, the specified request timeout action is executed.

Minimum value: 0

Maximum value: 86400

adptTimeout

Adapts the configured request timeout based on flow conditions. The timeout is increased or decreased internally and applied on the flow.

Possible values: ENABLED, DISABLED

Default value: DISABLED

reqTimeoutAction

Action to take when the HTTP request does not complete within the specified request timeout duration. You can configure the following actions:

* RESET - Send RST (reset) to client when timeout occurs.

* DROP - Drop silently when timeout occurs.

* Custom responder action - Name of the responder action to trigger when timeout occurs, used to send custom message.

dropExtraData

Drop any extra data when server sends more data than the specified content-length.

Possible values: ENABLED, DISABLED

Default value: DISABLED

webLog

Enable or disable web logging.

Possible values: ENABLED, DISABLED

Default value: ENABLED

clientIpHdrExpr

Name of the header that contains the real client IP address.

maxReq

Maximum requests allowed on a single connection.

Default value: 0

Minimum value: 0

Maximum value: 65534

persistentETag

Generate the persistent NetScaler specific ETag for the HTTP response with ETag header.

Possible values: ENABLED, DISABLED

Default value: DISABLED

spdy

Enable SPDYv2 or SPDYv3 or both over SSL vserver. SSL will advertise SPDY support during NPN Handshake. Both SPDY versions are enabled when this parameter is set to BOTH.

Possible values: DISABLED, ENABLED, V2, V3

Default value: DISABLED

reusePoolTimeout

Idle timeout (in seconds) for server connections in re-use pool. Connections in the re-use pool are flushed, if they remain idle for the configured timeout.

Default value: 0

Minimum value: 0

Maximum value: 31536000

maxHeaderLen

Number of bytes to be queued to look for complete header before returning error. If complete header is not obtained after queuing these many bytes, request will be marked as invalid and no L7 processing will be done for that TCP connection.

Default value: 24820

Minimum value: 2048

Maximum value: 61440

Example

```
set httpprofile <profile name> -dropInvalReqs ON -markHttp09Inval ON
```

unset ns httpProfile

Removes the attributes of the HTTP profile. Attributes for which a default value is available revert to their default values. Refer to the 'set ns httpProfile' command for a description of the parameters..Refer to the set ns httpProfile command for meanings of the arguments.

Synopsys

```
unset ns httpProfile <name> [-dropInvalReqs] [-markHttp09Inval] [-markConnReqInval] [-cmpOnPush] [-conMultiplex] [-maxReusePool] [-dropExtraCRLF] [-incompHdrDelay] [-webSocket] [-dropExtraData] [-clientIpHdrExpr] [-reqTimeout] [-adptTimeout] [-reqTimeoutAction] [-webLog] [-maxReq] [-persistentETag] [-spdy] [-reusePoolTimeout] [-maxHeaderLen] [-rtspTunnel]
```

show ns httpProfile

Displays information about HTTP profiles configured on the appliance.

Synopsys

```
show ns httpProfile [<name>]
```

Arguments

name

Name of the HTTP profile to be displayed. If a name is not provided, information about all HTTP profiles is shown.

Outputs

dropInvalReqs

Dropping of invalid HTTP requests/responses

markHttp09Inval

Invalidating HTTP 0.9 requests

markConnReqInval

Invalidating CONNECT HTTP requests

cmpOnPush

Compression on PUSH packet

conMultiplex

Reuse server connections for requests from more than one client connections.

maxReusePool

Maximum connections in reusepool

webSocket

HTTP connection to be upgraded to a web socket connection. Once upgraded, NetScaler does not process Layer 7 traffic on this connection.

refCnt

Number of entities using this profile

stateflag

State flag

dropExtraCRLF

Drop any extra 'CR' and 'LF' characters present after the header.

incompHdrDelay

Maximum time to wait, in milliseconds, between incomplete header packets. If the header packets take longer to arrive at NetScaler, the connection is silently dropped.

reqTimeout

Time, in seconds, within which the HTTP request must complete. If the request does not complete within this time, the specified request timeout action is executed.

adptTimeout

Adapts the configured request timeout based on flow conditions. The timeout is increased or decreased internally and applied on the flow.

reqTimeoutAction

Action to take when the HTTP request does not complete within the specified request timeout duration. You can configure the following actions:

- * RESET - Send RST (reset) to client when timeout occurs.

- * DROP - Drop silently when timeout occurs.

- * Custom responder action - Name of the responder action to trigger when timeout occurs, used to send custom message.

dropExtraData

Drop any extra data when server sends more data than the specified content-length.

webLog

Disabling weblog option

clientIpHdrExpr

Name of the header that contains the real client IP address.

maxReq

Maximum requests allowed on a single connection.

persistentETag

Generate the persistent NetScaler specific ETag for the HTTP response with ETag header.

spdy

Enable SPDYv2 or SPDYv3 or both over SSL vserver. SSL will advertise SPDY support during NPN Handshake. Both SPDY versions are enabled when this parameter is set to BOTH.

reusePoolTimeout

Idle timeout (in seconds) for server connections in re-use pool. Connections in the re-use pool are flushed, if they remain idle for the configured timeout.

maxHeaderLen

Number of bytes to be queued to look for complete header before returning error. If complete header is not obtained after queuing these many bytes, request will be marked as invalid and no L7 processing will be done for that TCP connection.

rtspTunnel

Allow RTSP tunnel in HTTP. Once application/x-rtsp-tunnelled is seen in Accept or Content-Type header, NetScaler does not process Layer 7 traffic on this connection.

devno**count**

Example

```
show http profile [profile name]
```

ns idletimeout

The following operations can be performed on "ns idletimeout":

[set](#) | [unset](#) | [show](#)

set ns idletimeout

Set the pcb/natpcb idletimeout. NOTE: This command is deprecated.

Synopsis

Arguments

tcpsvr

Set the idletimeout for server side pcb.

tcpclt

Set the idletimeout for client side pcb.

nontcpsvrclt

Set the idletimeout for natpcb.

Default value: 120

Minimum value: 1

Example

```
set ns idletimeout -tcpsvr 120  set ns idletimeout -tcpclt 120  set ns idletimeout -nontcpsvrclt 120
```

unset ns idletimeout

Use this command to remove ns idletimeout settings.Refer to the set ns idletimeout command for meanings of the arguments.NOTE: This command is deprecated.

Synopsis

show ns idletimeout

Display the global setting of pcb/natpcb idletimeout. NOTE: This command is deprecated.This command is deprecated in favour of 'set ns timeout'

Synopsis

Outputs

tcpsvr

Set the idletimeout for server side pcb.

tcpclt

Set the idletimeout for client side pcb.

nontcpsvrclt

Set the idletimeout for natpcb.

ns info

The following operations can be performed on "ns info":

show ns info

Displays the following details of the NetScaler appliance: * Software version * NetScaler IP address and subnet mask * Number of mapped IP addresses * Identifies the appliance as a standalone appliance, a part of an HA pair, or is a cluster node * Current time on the system and timestamp when the appliance was last updated * Features that are enabled or disabled * Modes that are enabled or disabled

Synopsys

show ns info

Example

An example of this command's output is shown below: System Rainier: Build 24, Date: Apr 2!

ns ip

The following operations can be performed on "ns ip":

add | **rm** | **set** | **unset** | **enable** | **disable** | **show**

add ns ip

Creates an IPv4 address on the NetScaler appliance.

Synopsys

```
add ns ip <IPAddress>@ <netmask> [-type <type> [-hostRoute ( ENABLED | DISABLED ) [-hostRtGw <ip_addr>] [-metric <integer>] [-vserverRHILevel <vserverRHILevel>] [-vserverRHIMode ( DYNAMIC_ROUTING | RISE )] [-ospfLSAType ( TYPE1 | TYPE5 ) [-ospfArea <positive_integer>]]] ] [-arp ( ENABLED | DISABLED )] [-icmp ( ENABLED | DISABLED )] [-vServer ( ENABLED | DISABLED )] [-telnet ( ENABLED | DISABLED )] [-ftp ( ENABLED | DISABLED )] [-gui <gui>] [-ssh ( ENABLED | DISABLED )] [-snmp ( ENABLED | DISABLED )] [-mgmtAccess ( ENABLED | DISABLED )] [-restrictAccess ( ENABLED | DISABLED )] [-dynamicRouting ( ENABLED | DISABLED )] [-state ( ENABLED | DISABLED )] [-vriD <positive_integer>] [-icmpResponse <icmpResponse>] [-ownerNode <positive_integer>] [-arpResponse <arpResponse>] [-td <positive_integer>]
```

Arguments

IPAddress

IPv4 address to create on the NetScaler appliance. Cannot be changed after the IP address is created.

netmask

Subnet mask associated with the IP address.

type

Type of the IP address to create on the NetScaler appliance. Cannot be changed after the IP address is created. The following are the different types of NetScaler owned IP addresses:

* A Subnet IP (SNIP) address is used by the NetScaler ADC to communicate with the servers. The NetScaler also uses the subnet IP address when generating its own packets, such as packets related to dynamic routing protocols, or to send monitor probes to check the health of the servers.

* A Virtual IP (VIP) address is the IP address associated with a virtual server. It is the IP address to which clients connect. An appliance managing a wide range of traffic may have many VIPs configured. Some of the attributes of the VIP address are customized to meet the requirements of the virtual server.

* A GSLB site IP (GSLBIP) address is associated with a GSLB site. It is not mandatory to specify a GSLBIP address when you initially configure the NetScaler appliance. A GSLBIP address is used only when you create a GSLB site.

* A Cluster IP (CLIP) address is the management address of the cluster. All cluster configurations must be performed by accessing the cluster through this IP address.

Possible values: SNIP, VIP, NSIP, GSLBsiteIP, CLIP

Default value: SNIP

arp

Respond to ARP requests for this IP address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

icmp

Respond to ICMP requests for this IP address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

vServer

Use this option to set (enable or disable) the virtual server attribute for this IP address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

telnet

Allow Telnet access to this IP address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

ftp

Allow File Transfer Protocol (FTP) access to this IP address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

gui

Allow graphical user interface (GUI) access to this IP address.

Possible values: ENABLED, SECUREONLY, DISABLED

Default value: ENABLED

ssh

Allow secure shell (SSH) access to this IP address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

snmp

Allow Simple Network Management Protocol (SNMP) access to this IP address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

mgmtAccess

Allow access to management applications on this IP address.

Possible values: ENABLED, DISABLED

Default value: DISABLED

restrictAccess

Block access to nonmanagement applications on this IP. This option is applicable for MIPs, SNIPs, and NSIP, and is disabled by default. Nonmanagement applications can run on the underlying NetScaler Free BSD operating system.

Possible values: ENABLED, DISABLED

Default value: DISABLED

dynamicRouting

Allow dynamic routing on this IP address. Specific to Subnet IP (SNIP) address.

Possible values: ENABLED, DISABLED

Default value: DISABLED

hostRoute

Advertise a route for the VIP address using the dynamic routing protocols running on the NetScaler appliance.

Possible values: ENABLED, DISABLED

hostRtGw

IP address of the gateway of the route for this VIP address.

Default value: -1

metric

Integer value to add to or subtract from the cost of the route advertised for the VIP address.

Minimum value: -16777215

vserverRHILevel

Advertise the route for the Virtual IP (VIP) address on the basis of the state of the virtual servers associated with that VIP.

* NONE - Advertise the route for the VIP address, regardless of the state of the virtual servers associated with the address.

* ONE VSERVER - Advertise the route for the VIP address if at least one of the associated virtual servers is in UP state.

* ALL VSERVER - Advertise the route for the VIP address if all of the associated virtual servers are in UP state.

* VSVR_CNTRL - Advertise the route for the VIP address according to the RHILevel (RHI STATE) parameter setting on all the associated virtual servers of the VIP address along with their states.

When Vserver RHI Level (RHI) parameter is set to VSVR_CNTRL, the following are different RHI behaviors for the VIP address on the basis of RHILevel (RHI STATE) settings on the virtual servers associated with the VIP address:

* If you set RHI STATE to PASSIVE on all virtual servers, the NetScaler ADC always advertises the route for the VIP address.

* If you set RHI STATE to ACTIVE on all virtual servers, the NetScaler ADC advertises the route for the VIP address if at least one of the associated virtual servers is in UP state.

* If you set RHI STATE to ACTIVE on some and PASSIVE on others, the NetScaler ADC advertises the route for the VIP address if at least one of the associated virtual servers, whose RHI STATE set to ACTIVE, is in UP state.

Possible values: ONE_VSERVER, ALL_VSERVERS, NONE, VSVR_CNTRL

Default value: ONE_VSERVER

vserverRHIMode

Advertise the route for the Virtual IP (VIP) address using dynamic routing protocols or using RISE

* DYNMAIC_ROUTING - Advertise the route for the VIP address using dynamic routing protocols (default)

* RISE - Advertise the route for the VIP address using RISE.

Possible values: DYNAMIC_ROUTING, RISE

Default value: DYNAMIC_ROUTING

ospfLSAType

Type of LSAs to be used by the OSPF protocol, running on the NetScaler appliance, for advertising the route for this VIP address.

Possible values: TYPE1, TYPE5

Default value: DISABLED

ospfArea

ID of the area in which the type1 link-state advertisements (LSAs) are to be advertised for this virtual IP (VIP) address by the OSPF protocol running on the NetScaler appliance. When this parameter is not set, the VIP is advertised on all areas.

Default value: -1

Minimum value: 0

Maximum value: 4294967294LU

state

Enable or disable the IP address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

vrID

A positive integer that uniquely identifies a VMAC address for binding to this VIP address. This binding is used to set up NetScaler appliances in an active-active configuration using VRRP.

Minimum value: 1

Maximum value: 255

icmpResponse

Respond to ICMP requests for a Virtual IP (VIP) address on the basis of the states of the virtual servers associated with that VIP. Available settings function as follows:

* NONE - The NetScaler appliance responds to any ICMP request for the VIP address, irrespective of the states of the virtual servers associated with the address.

* ONE VSERVER - The NetScaler appliance responds to any ICMP request for the VIP address if at least one of the associated virtual servers is in UP state.

* ALL VSERVER - The NetScaler appliance responds to any ICMP request for the VIP address if all of the associated virtual servers are in UP state.

* VSVR_CNTRL - The behavior depends on the ICMP VSERVER RESPONSE setting on all the associated virtual servers.

The following settings can be made for the ICMP VSERVER RESPONSE parameter on a virtual server:

* If you set ICMP VSERVER RESPONSE to PASSIVE on all virtual servers, NetScaler always responds.

* If you set ICMP VSERVER RESPONSE to ACTIVE on all virtual servers, NetScaler responds if even one virtual server is UP.

* When you set ICMP VSERVER RESPONSE to ACTIVE on some and PASSIVE on others, NetScaler responds if even one virtual server set to ACTIVE is UP.

Possible values: NONE, ONE_VSERVER, ALL_VSERVERS, VSVR_CNTRL

Default value: 5

ownerNode

The owner node in a Cluster for this IP address. Owner node can vary from 0 to 31. If ownernode is not specified then the IP is treated as Striped IP.

Default value: 255

Minimum value: 0

arpResponse

Respond to ARP requests for a Virtual IP (VIP) address on the basis of the states of the virtual servers associated with that VIP. Available settings function as follows:

- * NONE - The NetScaler appliance responds to any ARP request for the VIP address, irrespective of the states of the virtual servers associated with the address.
- * ONE VSERVER - The NetScaler appliance responds to any ARP request for the VIP address if at least one of the associated virtual servers is in UP state.
- * ALL VSERVER - The NetScaler appliance responds to any ARP request for the VIP address if all of the associated virtual servers are in UP state.

Possible values: NONE, ONE_VSERVER, ALL_VSERVERS

Default value: 5

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

Example

```
add ns ip 10.102.4.123 255.255.255.0
```

rm ns ip

Removes an IPv4 address configured on the NetScaler appliance.

Synopsis

```
rm ns ip <IPAddress>@ [-td <positive_integer>]
```

Arguments

IPAddress

IPv4 address that you want to remove.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

Example

```
rm ns ip 10.102.4.123
```

set ns ip

Modifies the parameters of an IPv4 address configured on the NetScaler appliance.

Synopsys

```
set ns ip (<IPAddress>@ [-td <positive_integer>]) [-netmask <netmask>] [-arp ( ENABLED | DISABLED )] [-icmp (
ENABLED | DISABLED )] [-vServer ( ENABLED | DISABLED )] [-telnet ( ENABLED | DISABLED )] [-ftp ( ENABLED |
DISABLED )] [-gui <gui>] [-ssh ( ENABLED | DISABLED )] [-snmp ( ENABLED | DISABLED )] [-mgmtAccess (
ENABLED | DISABLED )] [-restrictAccess ( ENABLED | DISABLED )] [-dynamicRouting ( ENABLED | DISABLED )] [-
hostRoute ( ENABLED | DISABLED )] [-hostRtGw <ip_addr>] [-metric <integer>] [-vserverRHILevel
<vserverRHILevel>] [-vserverRHIMode ( DYNAMIC_ROUTING | RISE )] [-ospfLSAType ( TYPE1 | TYPE5 )] [-
ospfArea <positive_integer>]]] [-vrID <positive_integer>] [-icmpResponse <icmpResponse>] [-arpResponse
<arpResponse>]
```

Arguments

IPAddress

IPv4 address whose parameters you want to modify.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

netmask

Subnet mask associated with the IP address.

arp

Respond to ARP requests for this IP address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

icmp

Respond to ICMP requests for this IP address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

vServer

Use this option to set (enable or disable) the virtual server attribute for this IP address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

telnet

Allow Telnet access to this IP address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

ftp

Allow File Transfer Protocol (FTP) access to this IP address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

gui

Allow graphical user interface (GUI) access to this IP address.

Possible values: ENABLED, SECUREONLY, DISABLED

Default value: ENABLED

ssh

Allow secure shell (SSH) access to this IP address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

snmp

Allow Simple Network Management Protocol (SNMP) access to this IP address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

mgmtAccess

Allow access to management applications on this IP address.

Possible values: ENABLED, DISABLED

Default value: DISABLED

restrictAccess

Block access to nonmanagement applications on this IP. This option is applicable for MIPs, SNIPs, and NSIP, and is disabled by default. Nonmanagement applications can run on the underlying NetScaler Free BSD operating system.

Possible values: ENABLED, DISABLED

Default value: DISABLED

dynamicRouting

Allow dynamic routing on this IP address. Specific to Subnet IP (SNIP) address.

Possible values: ENABLED, DISABLED

Default value: DISABLED

hostRoute

Advertise a route for the VIP address using the dynamic routing protocols running on the NetScaler appliance.

Possible values: ENABLED, DISABLED

hostRtGw

IP address of the gateway of the route for this VIP address.

Default value: -1

metric

Integer value to add to or subtract from the cost of the route advertised for the VIP address.

Minimum value: -16777215

vserverRHILevel

Advertise the route for the Virtual IP (VIP) address on the basis of the state of the virtual servers associated with that VIP.

- * NONE - Advertise the route for the VIP address, regardless of the state of the virtual servers associated with the address.

- * ONE VSERVER - Advertise the route for the VIP address if at least one of the associated virtual servers is in UP state.

- * ALL VSERVER - Advertise the route for the VIP address if all of the associated virtual servers are in UP state.

- * VSVR_CNTRL - Advertise the route for the VIP address according to the RHIstate (RHI STATE) parameter setting on all the associated virtual servers of the VIP address along with their states.

When Vserver RHI Level (RHI) parameter is set to VSVR_CNTRL, the following are different RHI behaviors for the VIP address on the basis of RHIstate (RHI STATE) settings on the virtual servers associated with the VIP address:

- * If you set RHI STATE to PASSIVE on all virtual servers, the NetScaler ADC always advertises the route for the VIP address.

- * If you set RHI STATE to ACTIVE on all virtual servers, the NetScaler ADC advertises the route for the VIP address if at least one of the associated virtual servers is in UP state.

- * If you set RHI STATE to ACTIVE on some and PASSIVE on others, the NetScaler ADC advertises the route for the VIP address if at least one of the associated virtual servers, whose RHI STATE set to ACTIVE, is in UP state.

Possible values: ONE_VSERVER, ALL_VSERVERS, NONE, VSVR_CNTRL

Default value: ONE_VSERVER

vserverRHIMode

Advertise the route for the Virtual IP (VIP) address using dynamic routing protocols or using RISE

- * DYNAMIC_ROUTING - Advertise the route for the VIP address using dynamic routing protocols (default)

- * RISE - Advertise the route for the VIP address using RISE.

Possible values: DYNAMIC_ROUTING, RISE

Default value: DYNAMIC_ROUTING

ospfLSAType

Type of LSAs to be used by the OSPF protocol, running on the NetScaler appliance, for advertising the route for this VIP address.

Possible values: TYPE1, TYPE5

Default value: DISABLED

ospfArea

ID of the area in which the type1 link-state advertisements (LSAs) are to be advertised for this virtual IP (VIP) address by the OSPF protocol running on the NetScaler appliance. When this parameter is not set, the VIP is advertised on all areas.

Default value: -1

Minimum value: 0

Maximum value: 4294967294LU

vrID

A positive integer that uniquely identifies a VMAC address for binding to this VIP address. This binding is used to set up NetScaler appliances in an active-active configuration using VRRP.

Minimum value: 1

Maximum value: 255

icmpResponse

Respond to ICMP requests for a Virtual IP (VIP) address on the basis of the states of the virtual servers associated with that VIP. Available settings function as follows:

- * NONE - The NetScaler appliance responds to any ICMP request for the VIP address, irrespective of the states of the virtual servers associated with the address.
- * ONE VSERVER - The NetScaler appliance responds to any ICMP request for the VIP address if at least one of the associated virtual servers is in UP state.
- * ALL VSERVER - The NetScaler appliance responds to any ICMP request for the VIP address if all of the associated virtual servers are in UP state.
- * VSVR_CNTRL - The behavior depends on the ICMP VSERVER RESPONSE setting on all the associated virtual servers.

The following settings can be made for the ICMP VSERVER RESPONSE parameter on a virtual server:

- * If you set ICMP VSERVER RESPONSE to PASSIVE on all virtual servers, NetScaler always responds.
- * If you set ICMP VSERVER RESPONSE to ACTIVE on all virtual servers, NetScaler responds if even one virtual server is UP.
- * When you set ICMP VSERVER RESPONSE to ACTIVE on some and PASSIVE on others, NetScaler responds if even one virtual server set to ACTIVE is UP.

Possible values: NONE, ONE_VSERVER, ALL_VSERVERS, VSVR_CNTRL

Default value: 5

arpResponse

Respond to ARP requests for a Virtual IP (VIP) address on the basis of the states of the virtual servers associated with that VIP. Available settings function as follows:

- * NONE - The NetScaler appliance responds to any ARP request for the VIP address, irrespective of the states of the virtual servers associated with the address.
- * ONE VSERVER - The NetScaler appliance responds to any ARP request for the VIP address if at least one of the associated virtual servers is in UP state.
- * ALL VSERVER - The NetScaler appliance responds to any ARP request for the VIP address if all of the associated virtual servers are in UP state.

Possible values: NONE, ONE_VSERVER, ALL_VSERVERS

Default value: 5

Example

```
set ns ip 10.102.4.123 -arp ENABLED
```

unset ns ip

Modifies the parameters of an IPv4 address configured on the NetScaler appliance..Refer to the set ns ip command for meanings of the arguments.

Synopsys

```
unset ns ip <IPAddress>@ [-td <positive_integer>] [-ospfArea] [-hostRtGw] [-netmask] [-arp] [-icmp] [-vServer] [-telnet] [-ftp] [-gui] [-ssh] [-snmp] [-mgmtAccess] [-restrictAccess] [-dynamicRouting] [-hostRoute] [-metric] [-vserverRHILevel] [-vserverRHIMode] [-ospfLSAType] [-vrID] [-icmpResponse] [-arpResponse]
```

Example

```
unset ns ip 10.102.4.123 -ospfArea
```

enable ns ip

Enables the specified VIP address configured on the NetScaler appliance.

Synopsys

```
enable ns ip (<IPAddress>@ [-td <positive_integer>])
```

Arguments

IPAddress

IP address that you want to enable.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

Example

```
enable ns ip 10.10.10.10
```

disable ns ip

Disables the specified VIP address configured on the NetScaler appliance.

Synopsys

```
disable ns ip (<IPAddress>@ [-td <positive_integer>])
```

Arguments

IPAddress

IP address that you want to disable.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

Example

```
disable ns ip 10.10.10.10
```

show ns ip

Displays settings of all the IPv4 addresses or of the specified IPv4 address configured on the NetScaler appliance. To display settings of all the IPv4 addresses, run the command without any parameters. To display settings of a particular IPv4 address, specify the IPv4 address.

Synopsys

show ns ip [<IPAddress> [-td <positive_integer>]] [-type <type>]

Arguments

IPAddress

IPv4 address whose details you want the NetScaler appliance to display.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

type

Display the settings of all IPv4 addresses of a particular type.

Possible values: SNIP, VIP, NSIP, GSLBsiteIP, CLIP

Default value: 0

Outputs

netmask

The netmask of this IP.

flags

The flags for this entry.

ipAttribute

arp

Whether arp is enabled or disabled.

icmp

Whether icmp is enabled or disabled.

vServer

Whether vserver is enabled or disabled.

telnet

Whether telnet is enabled or disabled.

ssh

Whether ssh is enabled or disabled.

gui

Whether gui is (enabled|SecureOnly|disabled).

snmp

Whether snmp is enabled or disabled.

ftp

Whether ftp is enabled or disabled.

mgmtAccess

Whether management access is enabled or disabled.

restrictAccess

Blocking of all ports not used for management access enabled or disabled

dynamicRouting

Whether dynamic routing is enabled or disabled.

bgp

Whether bgp is enabled or disabled.

ospf

Whether ospf is enabled or disabled.

rip

Whether rip is enabled or disabled.

hostRoute

Whether host route is enabled or disabled.

hostRtGw

Gateway used for advertising host route.

hostRtGwAct

Actual Gateway used for advertising host route.

metric

The metric value added or subtracted from the cost of the hostroute.

ospfArea

The area ID of the area in which OSPF Type1 LSAs are advertised. When ospfArea if not set, LSAs are advertised on all areas.

ospfAreaval

The area ID of the area in which OSPF Type1 LSAs are advertised.

vserverRHILevel

The rhi level for this IP.

vserverRHIMode

The rhi advertisement mode for this IP.

VIPrtadv2BSD

Whether this route is advertised to FreeBSD

VIPvserCount

Number of vservers bound to this VIP

VIPvserDownCount

Number of vservers bound to this VIP, which are down

VIPvsrvrRHIActiveCount

Number of vservers that have RHI state ACTIVE

VIPvsrvrRHIActiveUpCount

Number of vservers that have RHI state ACTIVE, which are UP

ospfLSAType

The ospf lsa type to use while advertising this IP.

state

Whether this ip is enabled or disabled.

freePorts

Number of free Ports available on this IP

vrID

A positive integer that uniquely identifies a VMAC address for binding to this VIP address. This binding is used to set up NetScaler appliances in an active-active configuration using VRRP.

riseRhiMsgCode

The code indicating the rise rhi status.

ipType**icmpResponse**

Respond to ICMP requests for a Virtual IP (VIP) address on the basis of the states of the virtual servers associated with that VIP. Available settings function as follows:

- * NONE - The NetScaler appliance responds to any ICMP request for the VIP address, irrespective of the states of the virtual servers associated with the address.
- * ONE VSERVER - The NetScaler appliance responds to any ICMP request for the VIP address if at least one of the associated virtual servers is in UP state.
- * ALL VSERVER - The NetScaler appliance responds to any ICMP request for the VIP address if all of the associated virtual servers are in UP state.
- * VSVR_CNTRL - The behavior depends on the ICMP VSERVER RESPONSE setting on all the associated virtual servers.

The following settings can be made for the ICMP VSERVER RESPONSE parameter on a virtual server:

- * If you set ICMP VSERVER RESPONSE to PASSIVE on all virtual servers, NetScaler always responds.
- * If you set ICMP VSERVER RESPONSE to ACTIVE on all virtual servers, NetScaler responds if even one virtual server is UP.
- * When you set ICMP VSERVER RESPONSE to ACTIVE on some and PASSIVE on others, NetScaler responds if even one virtual server set to ACTIVE is UP.

ownerNode

The owner node in a Cluster for this IP address. Owner node can vary from 0 to 31. If ownernode is not specified then the IP is treated as Striped IP.

arpResponse

Respond to ARP requests for a Virtual IP (VIP) address on the basis of the states of the virtual servers associated with that VIP. Available settings function as follows:

- * NONE - The NetScaler appliance responds to any ARP request for the VIP address, irrespective of the states of the virtual servers associated with the address.

* ONE VSERVER - The NetScaler appliance responds to any ARP request for the VIP address if at least one of the associated virtual servers is in UP state.

* ALL VSERVER - The NetScaler appliance responds to any ARP request for the VIP address if all of the associated virtual servers are in UP state.

stateflag

cfgflags

This contains the flags for IP in DB

ipRefCount

Used to keep reference count of IP

devno

count

Example

```
show ns ip Ipaddress      Type    Mode Arp  Icmp Vserver State  Owner -----  ----  ---
```

ns ip6

The following operations can be performed on "ns ip6":

add | **rm** | **set** | **unset** | **show**

add ns ip6

Creates an IPv6 address on the NetScaler appliance.

Synopsys

```
add ns ip6 <IPv6Address>@ [-scope ( global | link-local )] [-type <type>] [-hostRoute ( ENABLED | DISABLED )] [-ip6hostRtGw <ipv6_addr|*>] [-metric <integer>] [-vserverRHILevel <vserverRHILevel>] [-ospf6LSAType ( INTRA_AREA | EXTERNAL ) [-ospfArea <positive_integer>]]] [-vlan <positive_integer>] [-nd ( ENABLED | DISABLED )] [-icmp ( ENABLED | DISABLED )] [-vServer ( ENABLED | DISABLED )] [-telnet ( ENABLED | DISABLED )] [-ftp ( ENABLED | DISABLED )] [-gui <gui>] [-ssh ( ENABLED | DISABLED )] [-snmp ( ENABLED | DISABLED )] [-mgmtAccess ( ENABLED | DISABLED )] [-restrictAccess ( ENABLED | DISABLED )] [-dynamicRouting ( ENABLED | DISABLED )] [-state ( DISABLED | ENABLED )] [-map <ip_addr>] [-ownerNode <positive_integer>] [-td <positive_integer>]
```

Arguments

IPv6Address

IPv6 address to create on the NetScaler appliance.

scope

Scope of the IPv6 address to be created. Cannot be changed after the IP address is created.

Possible values: global, link-local

Default value: global

type

Type of IP address to be created on the NetScaler appliance. Cannot be changed after the IP address is created.

Possible values: NSIP, VIP, SNIP, GSLBsiteIP, ADNSsvclIP, CLIP

Default value: SNIP

vlan

The VLAN number.

Default value: 0

Minimum value: 0

Maximum value: 4094

nd

Respond to Neighbor Discovery (ND) requests for this IP address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

icmp

Respond to ICMP requests for this IP address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

vServer

Enable or disable the state of all the virtual servers associated with this VIP6 address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

telnet

Allow Telnet access to this IP address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

ftp

Allow File Transfer Protocol (FTP) access to this IP address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

gui

Allow graphical user interface (GUI) access to this IP address.

Possible values: ENABLED, SECUREONLY, DISABLED

Default value: ENABLED

ssh

Allow secure Shell (SSH) access to this IP address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

snmp

Allow Simple Network Management Protocol (SNMP) access to this IP address.

Possible values: ENABLED, DISABLED

Default value: ENABLED

mgmtAccess

Allow access to management applications on this IP address.

Possible values: ENABLED, DISABLED

Default value: DISABLED

restrictAccess

Block access to nonmanagement applications on this IP address. This option is applicable for MIP6s, SNIP6s, and NSIP6s, and is disabled by default. Nonmanagement applications can run on the underlying NetScaler Free BSD operating system.

Possible values: ENABLED, DISABLED

Default value: DISABLED

dynamicRouting

Allow dynamic routing on this IP address. Specific to Subnet IPv6 (SNIP6) address.

Possible values: ENABLED, DISABLED

Default value: DISABLED

hostRoute

Advertise a route for the VIP6 address by using the dynamic routing protocols running on the NetScaler appliance.

Possible values: ENABLED, DISABLED

ip6hostRtGw

IPv6 address of the gateway for the route. If Gateway is not set, VIP uses :: as the gateway.

Default value: 0

metric

Integer value to add to or subtract from the cost of the route advertised for the VIP6 address.

Minimum value: -16777215

vserverRHILevel

Advertise or do not advertise the route for the Virtual IP (VIP6) address on the basis of the state of the virtual servers associated with that VIP6.

- * NONE - Advertise the route for the VIP6 address, irrespective of the state of the virtual servers associated with the address.
- * ONE VSERVER - Advertise the route for the VIP6 address if at least one of the associated virtual servers is in UP state.
- * ALL VSERVER - Advertise the route for the VIP6 address if all of the associated virtual servers are in UP state.
- * VSVR_CNTRL. Advertise the route for the VIP address according to the RHILevel (RHI STATE) parameter setting on all the associated virtual servers of the VIP address along with their states.

When Vserver RHI Level (RHI) parameter is set to VSVR_CNTRL, the following are different RHI behaviors for the VIP address on the basis of RHILevel (RHI STATE) settings on the virtual servers associated with the VIP address:

- * If you set RHI STATE to PASSIVE on all virtual servers, the NetScaler ADC always advertises the route for the VIP address.
- * If you set RHI STATE to ACTIVE on all virtual servers, the NetScaler ADC advertises the route for the VIP address if at least one of the associated virtual servers is in UP state.
- * If you set RHI STATE to ACTIVE on some and PASSIVE on others, the NetScaler ADC advertises the route for the VIP address if at least one of the associated virtual servers, whose RHI STATE set to ACTIVE, is in UP state.

Possible values: ONE_VSERVER, ALL_VSERVERS, NONE, VSVR_CNTRL

Default value: ONE_VSERVER

ospf6LSAType

Type of LSAs to be used by the IPv6 OSPF protocol, running on the NetScaler appliance, for advertising the route for the VIP6 address.

Possible values: INTRA_AREA, EXTERNAL

Default value: EXTERNAL

ospfArea

ID of the area in which the Intra-Area-Prefix LSAs are to be advertised for the VIP6 address by the IPv6 OSPF protocol running on the NetScaler appliance. When ospfArea is not set, VIP6 is advertised on all areas.

Default value: -1

Minimum value: 0

Maximum value: 4294967294LU

state

Enable or disable the IP address.

Possible values: DISABLED, ENABLED

Default value: ENABLED

map

Mapped IPV4 address for the IPV6 address.

ownerNode

ID of the cluster node for which you are adding the IP address. Must be used if you want the IP address to be active only on the specific node. Can be configured only through the cluster IP address. Cannot be changed after the IP address is created.

Default value: 255

Minimum value: 0

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

Example

```
add ns ip6 2001::a/96 -scope GLOBAL
```

rm ns ip6

Removes an IPv6 address configured on the NetScaler appliance.

Synopsis

```
rm ns ip6 <IPv6Address>@ [-td <positive_integer>]
```

Arguments

IPv6Address

IPv6 address that you want to remove.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

Example

```
rm ns ip6 2002::5
```

set ns ip6

Modifies the specified parameters of an IPv6 address configured on the NetScaler appliance.

Synopsys

```
set ns ip6 (<IPv6Address>@ [-td <positive_integer>]) [-nd ( ENABLED | DISABLED )] [-icmp ( ENABLED |  
DISABLED )] [-vServer ( ENABLED | DISABLED )] [-telnet ( ENABLED | DISABLED )] [-ftp ( ENABLED | DISABLED  
)] [-gui <gui>] [-ssh ( ENABLED | DISABLED )] [-snmp ( ENABLED | DISABLED )] [-mgmtAccess ( ENABLED |  
DISABLED )] [-restrictAccess ( ENABLED | DISABLED )] [-state ( DISABLED | ENABLED )] [-map <ip_addr>] [-  
dynamicRouting ( ENABLED | DISABLED )] [-hostRoute ( ENABLED | DISABLED )] [-ip6hostRtGw <ipv6_addr|*>] [-  
metric <integer>] [-vserverRHILevel <vserverRHILevel>] [-ospf6LSAType ( INTRA_AREA | EXTERNAL )] [-ospfArea  
<positive_integer>]]]
```

Arguments

IPv6Address

IPv6 address whose parameters you want to modify.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

nd

The state of ND responses for the entity.

Possible values: ENABLED, DISABLED

Default value: ENABLED

icmp

The state of ICMP responses for the entity.

Possible values: ENABLED, DISABLED

Default value: ENABLED

vServer

The state of vserver attribute for this IP entity.

Possible values: ENABLED, DISABLED

Default value: ENABLED

telnet

The state of telnet access to this IP entity.

Possible values: ENABLED, DISABLED

Default value: ENABLED

ftp

The state of ftp access to this IP entity.

Possible values: ENABLED, DISABLED

Default value: ENABLED

gui

The state of GUI access to this IP entity.

Possible values: ENABLED, SECUREONLY, DISABLED

Default value: ENABLED

ssh

The state of SSH access to this IP entity.

Possible values: ENABLED, DISABLED

Default value: ENABLED

snmp

The state of SNMP access to this IP entity.

Possible values: ENABLED, DISABLED

Default value: ENABLED

mgmtAccess

The state of management access to this IP entity.

Possible values: ENABLED, DISABLED

Default value: DISABLED

restrictAccess

Status of ports not used for management access (blocked/open) for the entity.

Possible values: ENABLED, DISABLED

Default value: DISABLED

state

Enable or disable the IP address.

Possible values: DISABLED, ENABLED

Default value: ENABLED

map

Mapped IPV4 address for the IPV6 address.

dynamicRouting

Allow dynamic routing on this IP address. Specific to Subnet IPv6 (SNIP6) address.

Possible values: ENABLED, DISABLED

Default value: DISABLED

hostRoute

Advertise a route for the VIP6 address by using the dynamic routing protocols running on the NetScaler appliance.

Possible values: ENABLED, DISABLED

ip6hostRtGw

IPv6 address of the gateway for the route. If Gateway is not set, VIP uses :: as the gateway.

Default value: 0

metric

Integer value to add to or subtract from the cost of the route advertised for the VIP6 address.

Minimum value: -16777215

vserverRHILevel

Advertise or do not advertise the route for the Virtual IP (VIP6) address on the basis of the state of the virtual servers associated with that VIP6.

- * NONE - Advertise the route for the VIP6 address, irrespective of the state of the virtual servers associated with the address.
- * ONE VSERVER - Advertise the route for the VIP6 address if at least one of the associated virtual servers is in UP state.
- * ALL VSERVER - Advertise the route for the VIP6 address if all of the associated virtual servers are in UP state.
- * VSVR_CNTRL - Advertise the route for the VIP address according to the RHILevel (RHI STATE) parameter setting on all the associated virtual servers of the VIP address along with their states.

When Vserver RHI Level (RHI) parameter is set to VSVR_CNTRL, the following are different RHI behaviors for the VIP address on the basis of RHILevel (RHI STATE) settings on the virtual servers associated with the VIP address:

- * If you set RHI STATE to PASSIVE on all virtual servers, the NetScaler ADC always advertises the route for the VIP address.
- * If you set RHI STATE to ACTIVE on all virtual servers, the NetScaler ADC advertises the route for the VIP address if at least one of the associated virtual servers is in UP state.
- * If you set RHI STATE to ACTIVE on some and PASSIVE on others, the NetScaler ADC advertises the route for the VIP address if at least one of the associated virtual servers, whose RHI STATE set to ACTIVE, is in UP state.

Possible values: ONE_VSERVER, ALL_VSERVERS, NONE, VSVR_CNTRL

Default value: ONE_VSERVER

ospf6LSAType

The OSPF's route advertisement type.

Possible values: INTRA_AREA, EXTERNAL

Default value: EXTERNAL

ospfArea

ID of the area in which the Intra-Area-Prefix LSAs are to be advertised for the VIP6 address by the IPv6 OSPF protocol running on the NetScaler appliance. When ospfArea is not set, VIP6 is advertised on all areas.

Default value: -1

Minimum value: 0

Maximum value: 4294967294LU

Example

```
set ns ip6 2001::a -map 10.102.33.27
```

unset ns ip6

Modifies the parameters of an IPv6 address configured on the NetScaler appliance..Refer to the set ns ip6 command for meanings of the arguments.

Synopsys

```
unset ns ip6 <IPv6Address>@ [-td <positive_integer>] [-ospfArea] [-nd] [-icmp] [-vServer] [-telnet] [-ftp] [-gui] [-ssh] [-snmp] [-mgmtAccess] [-restrictAccess] [-state] [-map] [-dynamicRouting] [-hostRoute] [-ip6hostRtGw] [-metric] [-vserverRHILevel] [-ospf6LSAType]
```

Example

```
unset ns ip6 2001::a -ospfArea
```

show ns ip6

Displays settings of all the IPv6 addresses or of the specified IPv6 address configured on the NetScaler appliance. To display settings of all the IPv6 addresses, run the command without any parameters. To display settings of a particular IPv6 address, specify the IPv6 address.

Synopsys

```
show ns ip6 [<IPv6Address> [-td <positive_integer>]]
```

Arguments

IPv6Address

IPv6 address whose settings you want the NetScaler appliance to display.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

Outputs

scope

Scope of the IPv6 address to be created. Cannot be changed after the IP address is created.

type

The type of the IPV6 address

ipType

The type of the IPv6 address

vlan

The VLAN number.

nd

Whether ND is enabled or disabled.

icmp

Whether icmp is enabled or disabled.

vServer

Whether vsrver is enabled or disabled.

telnet

Whether telnet is enabled or disabled.

ssh

Whether ssh is enabled or disabled.

gui

Whether gui is (enabled|SecureOnly|disabled).

snmp

Whether snmp is enabled or disabled.

ftp

Whether ftp is enabled or disabled.

mgmtAccess

Whether management access is enabled or disabled.

restrictAccess

Blocking of all ports not used for management access enabled or disabled

state

Current state of this IP.

map

Mapped IPV4 address for the IPV6 address.

dynamicRouting

Allow dynamic routing on this IP address. Specific to Subnet IPv6 (SNIP6) address.

hostRoute

Advertise a route for the VIP6 address by using the dynamic routing protocols running on the NetScaler appliance.

ip6hostRtGw

IPv6 address of the gateway for the route. If Gateway is not set, VIP uses :: as the gateway.

metric

The metric value to be added or subtracted from the cost of the hostroute advertised for this IPv6 entity.

vserverRHILevel

Advertise or do not advertise the route for the Virtual IP (VIP6) address on the basis of the state of the virtual servers associated with that VIP6.

* NONE - Advertise the route for the VIP6 address, irrespective of the state of the virtual servers associated with the address.

* ONE VSERVER - Advertise the route for the VIP6 address if at least one of the associated virtual servers is in UP state.

* ALL VSERVER - Advertise the route for the VIP6 address if all of the associated virtual servers are in UP state.

* VSVR_CNTRL. Advertise the route for the VIP address according to the RHILstate (RHI STATE) parameter setting on all the associated virtual servers of the VIP address along with their states.

When Vserver RHI Level (RHI) parameter is set to VSVR_CNTRLD, the following are different RHI behaviors for the VIP address on the basis of RHlstate (RHI STATE) settings on the virtual servers associated with the VIP address:

- * If you set RHI STATE to PASSIVE on all virtual servers, the NetScaler ADC always advertises the route for the VIP address.

- * If you set RHI STATE to ACTIVE on all virtual servers, the NetScaler ADC advertises the route for the VIP address if at least one of the associated virtual servers is in UP state.

- * If you set RHI STATE to ACTIVE on some and PASSIVE on others, the NetScaler ADC advertises the route for the VIP address if at least one of the associated virtual servers, whose RHI STATE set to ACTIVE, is in UP state.

VIPrtadv2BSD

Whether this route is advertised to FreeBSD

VIPvserCount

Number of vservers bound to this VIP

VIPvserDownCount

Number of vservers bound to this VIP, which are down

ospf6LSAType

The OSPF's route advertisement type.

ospfArea

The area ID of the area in which OSPF INTRA AREA PREFIX LSAs should be advertised. When ospfArea is not set, LSAs are advertised in all areas.

ownerNode

ID of the cluster node for which you are adding the IP address. Must be used if you want the IP address to be active only on the specific node. Can be configured only through the cluster IP address. Cannot be changed after the IP address is created.

stateflag

cfgflags

This contains the flags for IP in DB

ipRefCount

Used to keep reference count of IPv6

systemType

The type of the System. Possible Values: Standalone, HA, Cluster. Used for display purpose.

devno

count

Example

```
show ns ip6
```

ns license

The following operations can be performed on "ns license":

show ns license

Displays the state of all the licensed features.

Synopsys

show ns license

Outputs

WL

Web Logging.

SP

Surge Protection.

LB

Load Balancing.

CS

Content Switching.

CR

Cache Redirect.

SC

Sure Connect.

CMP

Compression.

DELTA

Delta Compression.

PQ

Priority Queuing.

SSL

Secure Sockets Layer.

GSLB

Global Server Load Balancing.

GSLBP

GSLB Proximity.

HDOSP

DOS Protection.

Routing

Routing.

CF

Content Filter.

ContentAccelerator

transparent Integrated Caching.

IC

Integrated Caching.

SSLVPN

SSL VPN.

AAA

AAA

OSPF

OSPF Routing.

RIP

RIP Routing.

BGP

BGP Routing.

REWRITE

Rewrite.

IPv6PT

IPv6 protocol translation

AppFw

Application Firewall.

RESPONDER

Responder.

AGEE

NSXN

HTMLInjection

HTML Injection.

ModelID

Model Number ID.

push

NetScaler Push.

WlonNS

WI on NS.

AppFlow

AppFlow.

CloudBridge

CloudBridge.

CloudBridgeAppliance

CloudExtenderAppliance

ISIS

ISIS Routing.

Cluster

Clustering

CH

Call Home.

AppQoE

AppQoS

APPFLOWICA

Appflow for ICA

isStandardLic

Standard License

isEnterpriseLic

Enterprise License

isPlatinumLic

Platinum License

RISE

RISE

DiskCaching

Integrated Disk Cache

vPath

Vpath

FEO

Front End Optimization

ns limitIdentifier

The following operations can be performed on "ns limitIdentifier":

add | **rm** | **set** | **unset** | **show** | **stat**

add ns limitIdentifier

Adds a limit identifier to check if the amount of traffic exceeds a specified value, within a particular time interval.

Synopsys

```
add ns limitIdentifier <limitIdentifier> [-threshold <positive_integer>] [-timeSlice <positive_integer>] [-mode <mode> [-limitType ( BURSTY | SMOOTH )]] [-selectorName <string>] [-maxBandwidth <positive_integer>] [-trapsInTimeSlice <positive_integer>]
```

Arguments

limitIdentifier

Name for a rate limit identifier. Must begin with an ASCII letter or underscore (_) character, and must consist only of ASCII alphanumeric or underscore characters. Reserved words must not be used.

threshold

Maximum number of requests that are allowed in the given timeslice when requests (mode is set as REQUEST_RATE) are tracked per timeslice.

When connections (mode is set as CONNECTION) are tracked, it is the total number of connections that would be let through.

Default value: 1

Minimum value: 1

timeSlice

Time interval, in milliseconds, specified in multiples of 10, during which requests are tracked to check if they cross the threshold. This argument is needed only when the mode is set to REQUEST_RATE.

Default value: 1000

Minimum value: 10

mode

Defines the type of traffic to be tracked.

* REQUEST_RATE - Tracks requests/timeslice.

* CONNECTION - Tracks active transactions.

Examples

1. To permit 20 requests in 10 ms and 2 traps in 10 ms:

```
add limitIdentifier limit_req -mode request_rate -limitType smooth -timeslice 1000 -Threshold 2000 -trapsInTimeSlice 200
```

2. To permit 50 requests in 10 ms:

```
set limitIdentifier limit_req -mode request_rate -timeslice 1000 -Threshold 5000 -limitType smooth
```

3. To permit 1 request in 40 ms:

```
set limitIdentifier limit_req -mode request_rate -timeslice 2000 -Threshold 50 -limitType smooth
```

4. To permit 1 request in 200 ms and 1 trap in 130 ms:

```
set limitIdentifier limit_req -mode request_rate -timeslice 1000 -Threshold 5 -limitType smooth -trapsInTimeSlice 8
```

5. To permit 5000 requests in 1000 ms and 200 traps in 1000 ms:

```
set limitIdentifier limit_req -mode request_rate -timeslice 1000 -Threshold 5000 -limitType BURSTY
```

Possible values: CONNECTION, REQUEST_RATE, NONE

Default value: REQUEST_RATE

limitType

Smooth or bursty request type.

* SMOOTH - When you want the permitted number of requests in a given interval of time to be spread evenly across the timeslice

* BURSTY - When you want the permitted number of requests to exhaust the quota anytime within the timeslice.

This argument is needed only when the mode is set to REQUEST_RATE.

Possible values: BURSTY, SMOOTH

Default value: BURSTY

selectorName

Name of the rate limit selector. If this argument is NULL, rate limiting will be applied on all traffic received by the virtual server or the NetScaler (depending on whether the limit identifier is bound to a virtual server or globally) without any filtering.

maxBandwidth

Maximum bandwidth permitted, in kbps.

Minimum value: 0

Maximum value: 4294967287

trapsInTimeSlice

Number of traps to be sent in the timeslice configured. A value of 0 indicates that traps are disabled.

Minimum value: 0

Maximum value: 65535

Example

```
add ns limitIdentifier limit_id -threshold 2 -timeSlice 5000 -mode CONNECTION -selectorName
```

rm ns limitIdentifier

Removes a rate limit identifier from the appliance.

Synopsis

```
rm ns limitIdentifier <limitIdentifier>
```

Arguments

limitIdentifier

Name of the rate limit identifier to be removed.

Example

```
rm ns limitIdentifier limit_id
```

set ns limitIdentifier

Modifies the attributes of a rate limit identifier.

Synopsys

```
set ns limitIdentifier <limitIdentifier> [-threshold <positive_integer>] [-timeSlice <positive_integer>] [-mode <mode> [-limitType ( BURSTY | SMOOTH )]] [-selectorName <string>] [-maxBandwidth <positive_integer>] [-trapsInTimeSlice <positive_integer>]
```

Arguments

limitIdentifier

Name of the rate limit identifier to be modified.

threshold

Maximum number of requests that are allowed in the given timeslice when requests (mode is set as REQUEST_RATE) are tracked per timeslice.

When connections (mode is set as CONNECTION) are tracked, it is the total number of connections that would be let through.

Default value: 1

Minimum value: 1

timeSlice

Time interval, in milliseconds, specified in multiples of 10, during which requests are tracked to check if they cross the threshold. This argument is needed only when the mode is set to REQUEST_RATE.

Default value: 1000

Minimum value: 10

mode

Defines the type of traffic to be tracked.

* REQUEST_RATE - Tracks requests/timeslice.

* CONNECTION - Tracks active transactions.

Examples

1. To permit 20 requests in 10 ms and 2 traps in 10 ms:

```
add limitIdentifier limit_req -mode request_rate -limitType smooth -timeslice 1000 -Threshold 2000 -trapsInTimeSlice 200
```

2. To permit 50 requests in 10 ms:

```
set limitIdentifier limit_req -mode request_rate -timeslice 1000 -Threshold 5000 -limitType smooth
```

3. To permit 1 request in 40 ms:

```
set limitIdentifier limit_req -mode request_rate -timeslice 2000 -Threshold 50 -limitType smooth
```

4. To permit 1 request in 200 ms and 1 trap in 130 ms:

```
set limitIdentifier limit_req -mode request_rate -timeslice 1000 -Threshold 5 -limitType smooth -trapsInTimeSlice 8
```

5. To permit 5000 requests in 1000 ms and 200 traps in 1000 ms:

```
set limitIdentifier limit_req -mode request_rate -timeslice 1000 -Threshold 5000 -limitType BURSTY
```

Possible values: CONNECTION, REQUEST_RATE, NONE

Default value: REQUEST_RATE

limitType

Smooth or bursty request type.

* SMOOTH - When you want the permitted number of requests in a given interval of time to be spread evenly across the timeslice

* BURSTY - When you want the permitted number of requests to exhaust the quota anytime within the timeslice.

This argument is needed only when the mode is set to REQUEST_RATE.

Possible values: BURSTY, SMOOTH

Default value: BURSTY

selectorName

Name of the rate limit selector. If this argument is NULL, rate limiting will be applied on all traffic received by the virtual server or the NetScaler (depending on whether the limit identifier is bound to a virtual server or globally) without any filtering.

maxBandwidth

Maximum bandwidth permitted, in kbps.

Minimum value: 0

Maximum value: 4294967287

trapsInTimeSlice

Number of traps to be sent in the timeslice configured. A value of 0 indicates that traps are disabled.

Minimum value: 0

Maximum value: 65535

Example

```
set ns limitIdentifier limit_id -threshold 2 -timeSlice 5000 -mode CONNECTION -selectorName
```

unset ns limitIdentifier

Use this command to remove ns limitIdentifier settings. Refer to the set ns limitIdentifier command for meanings of the arguments.

Synopsis

```
unset ns limitIdentifier <limitIdentifier> [-selectorName] [-threshold] [-timeSlice] [-mode] [-limitType] [-maxBandwidth] [-trapsInTimeSlice]
```

show ns limitIdentifier

Displays information about a rate limit identifier.

Synopsis

```
show ns limitIdentifier [<limitIdentifier>]
```

Arguments

limitIdentifier

Name of the rate limit identifier about which to display information. If a name is not provided, information about all rate limit identifiers is shown.

Outputs

ngname

Nodegroup name to which this identifier belongs to.

threshold

Maximum number of requests that are allowed in the given timeslice when requests (mode is set as REQUEST_RATE) are tracked per timeslice.

When connections (mode is set as CONNECTION) are tracked, it is the total number of connections that would be let through.

timeSlice

Defines the time interval in msec specified in multiples of 10 msec during which the requests are tracked to see if they cross the threshold. It is used and displayed only when the mode is REQUEST_RATE while tracking request rate and for defining the trap timeslice.

mode

Defines the type of traffic to be tracked.

* REQUEST_RATE - Tracks requests/timeslice.

* CONNECTION - Tracks active transactions.

Examples

1. To permit 20 requests in 10 ms and 2 traps in 10 ms:

```
add limitidentifier limit_req -mode request_rate -limitType smooth -timeslice 1000 -Threshold 2000 -trapsInTimeSlice 200
```

2. To permit 50 requests in 10 ms:

```
set limitidentifier limit_req -mode request_rate -timeslice 1000 -Threshold 5000 -limitType smooth
```

3. To permit 1 request in 40 ms:

```
set limitidentifier limit_req -mode request_rate -timeslice 2000 -Threshold 50 -limitType smooth
```

4. To permit 1 request in 200 ms and 1 trap in 130 ms:

```
set limitidentifier limit_req -mode request_rate -timeslice 1000 -Threshold 5 -limitType smooth -trapsInTimeSlice 8
```

5. To permit 5000 requests in 1000 ms and 200 traps in 1000 ms:

```
set limitidentifier limit_req -mode request_rate -timeslice 1000 -Threshold 5000 -limitType BURSTY
```

limitType

Smooth or bursty request type.

* SMOOTH - When you want the permitted number of requests in a given interval of time to be spread evenly across the timeslice

* BURSTY - When you want the permitted number of requests to exhaust the quota anytime within the timeslice.

This argument is needed only when the mode is set to REQUEST_RATE.

selectorName

Name of the rate limit selector. If this argument is NULL, rate limiting will be applied on all traffic received by the virtual server or the NetScaler (depending on whether the limit identifier is bound to a virtual server or globally) without any filtering.

stateflag

This is used internally to identify ip addresses returned.

hits

The number of times this identifier was evaluated.

drop

The number of times action was taken.

rule

Rule.

time

Time interval considered for rate limiting

total

Maximum number of requests permitted in the computed timeslice

maxBandwidth

The maximum bandwidth in kbps permitted

trapsInTimeSlice

The maximum bandwidth permitted in kbps

trapsComputedInTimeSlice

The number of traps that would be sent in the timeslice configured.

computedTrapTimeSlice

The time interval computed for sending traps.

referenceCount

Total number of transactions pointing to this entry.

devno**count**

Example

```
show ns limitIdentifier limit_id
```

stat ns limitIdentifier

Display statistics of a identifier.

Synopsys

```
stat ns limitIdentifier [<name> [<pattern> ...]] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile  
<input_filename>] [-clearstats ( basic | full )] [-sortBy Hits [<sortOrder>]]
```

Arguments

name

The name of the identifier.

pattern

Pattern for the selector field, ? means field is required, * means field value does not matter, anything else is a regular pattern

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

sortBy

use this argument to sort by specific key

Possible values: Hits

sortOrder

use this argument to specify sort order

Possible values: ascending, descending

Default value: SORT_DESCENDING

Outputs

count

devno

stateflag

Outputs

Rate Limit Identifier Hits (Hits)

Total hits.

Rate Limit Identifier Drops (Drops)

Total drops

Rate Limit Session Hits (Hits)

Total hits.

ns limitSelector

The following operations can be performed on "ns limitSelector":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add ns limitSelector

NOTE: This command is deprecated.Replaced by "stream selector"

Synopsys

Arguments

selectorName

rule

rm ns limitSelector

NOTE: This command is deprecated.Replaced by "stream selector"

Synopsys

Arguments

selectorName

set ns limitSelector

NOTE: This command is deprecated.

Synopsys

Arguments

selectorName

rule

unset ns limitSelector

Use this command to remove ns limitSelector settings.Refer to the set ns limitSelector command for meanings of the arguments.NOTE: This command is deprecated.

Synopsys

show ns limitSelector

NOTE: This command is deprecated.Replaced by "stream selector"

Synopsys

Arguments

selectorName

Outputs

rule

stateflag

devno

count

ns limitSessions

The following operations can be performed on "ns limitSessions":

[show](#) | [clear](#)

show ns limitSessions

Displays the rate limit sessions available on the appliance.

Synopsys

show ns limitSessions <limitIdentifier> [-detail]

Arguments

limitIdentifier

Name of the rate limit identifier for which to display the sessions.

detail

Show the individual hash values.

Outputs

timeout

The time remaining on the session before a flush can be attempted. If active transactions are present the session will not be flushed

hits

The number of times this entry was hit.

drop

The number of times action was taken.

number

The hash of the matched selectlets.

name

The string formed by gathering selectlet values.

unit

Total computed hash of the matched selectlets.

flags

Used internally to identify ip addresses.

referenceCount

Total number of transactions pointing to this entry. Its the sum total of the connection and bandwidth references

maxBandwidth

The current bandwidth

SelectorIPv61

First IPV6 address gathered.

SelectorIPV62

Second IPV6 address gathered.

flag

Used internally to identify ipv6 addresses.

devno**count****stateflag**

clear ns limitSessions

Clears the rate limit sessions available on the appliance.

Synopsys

clear ns limitSessions <limitIdentifier>

Arguments

limitIdentifier

Name of the rate limit identifier for which the sessions must be cleared.

ns memory

The following operations can be performed on "ns memory":

stat ns memory

Displays memory statistics of NetScaler features.

Synopsys

```
stat ns memory [<pool>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full )]
```

Arguments

pool

Feature name for which to display memory statistics.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Maximum Memory Available (MB) (MemTotAvail)

Total system memory available for PE to grab from the system.

Allocation failure (AllocF)

Memory allocation failure for particular feature.

Percentage of memory allocated (Alloc(%))

Percentage of NetScaler memory used by the feature.

Total memory Allocated (KB) (CurAlloc(KB))

Total current NetScaler memory available for use by the feature, in kilobytes.

ns mode

The following operations can be performed on "ns mode":

[enable](#) | [disable](#) | [show](#)

enable ns mode

Enables NetScaler mode(s).

Synopsys

```
enable ns mode <Mode> ...
```

Arguments

Mode

Mode to be enabled. Multiple modes can be specified by providing a blank space between each mode.

Example

This CLI command enables the system's client keep-alive feature: `enable ns mode CKA`

disable ns mode

Disables NetScaler mode(s).

Synopsys

```
disable ns mode <Mode> ...
```

Arguments

Mode

Mode to be disabled. Multiple modes can be specified by providing a blank space between each mode.

Example

This example shows the command to disable the system's client keep-alive feature: `disable`

show ns mode

Displays the current state of NetScaler modes.

Synopsys

```
show ns mode
```

Outputs

Mode

Mode to be enabled. Multiple modes can be specified by providing a blank space between each mode.

FR

Fast Ramp.

L2

Layer 2 mode.

USIP

Use Source IP.

CKA

Client Keep-alive.

TCPB

TCP Buffering.

MBF

MAC-based forwarding.

Edge

Edge configuration.

USNIP

Use Subnet IP.

L3

Layer 3 mode (ip forwarding).

PMTUD

Path MTU Discovery.

SRADV

Static Route Advertisement.

DRADV

Direct Route Advertisement.

IRADV

Intranet Route Advertisement.

SRADV6

Ipv6 Static Route Advertisement.

DRADV6

Ipv6 Direct Route Advertisement.

BridgeBPDUs

BPDU Bridging Mode.

RISE_APBR

APBR Publishing Mode.

RISE_RHI

RHI Publishing Mode.

ns ns.conf

The following operations can be performed on "ns ns.conf":

show ns ns.conf

Displays the saved configurations.

Synopsys

show ns ns.conf

Outputs

textBlob

Text of the last saved configuration.

ns param

The following operations can be performed on "ns param":

[set](#) | [unset](#) | [show](#)

set ns param

Sets the parameters of the NetScaler appliance.

Synopsys

```
set ns param [-httpPort <port> ...] [-maxConn <positive_integer>] [-maxReq <positive_integer>] [-cip ( ENABLED | DISABLED ) <cipHeader>] [-cookieversion ( 0 | 1 )] [-secureCookie ( ENABLED | DISABLED )] [-pmtuMin <positive_integer>] [-pmtuTimeout <mins>] [-ftpPortRange <int[-int]>] [-crPortRange <int[-int]>] [-timezone <timezone>] [-grantQuotaMaxClient <positive_integer>] [-exclusiveQuotaMaxClient <positive_integer>] [-grantQuotaSpillOver <positive_integer>] [-exclusiveQuotaSpillOver <positive_integer>] [-useproxyport ( ENABLED | DISABLED )] [-internaluserlogin ( ENABLED | DISABLED )] [-aftpAllowRandomSourcePort ( ENABLED | DISABLED )] [-icaPorts <port> ...] [-tcpCIP ( ENABLED | DISABLED )]
```

Arguments

httpPort

HTTP ports on the web server. This allows the system to perform connection off-load for any client request that has a destination port matching one of these configured ports.

Minimum value: 1

Maximum value: 65535

maxConn

Maximum number of connections that will be made from the appliance to the web server(s) attached to it. The value entered here is applied globally to all attached servers.

Default value: 0

Minimum value: 0

Maximum value: 4294967294

maxReq

Maximum number of requests that the system can pass on a particular connection between the appliance and a server attached to it. Setting this value to 0 allows an unlimited number of requests to be passed. This value is overridden by the maximum number of requests configured on the individual service.

Minimum value: 0

Maximum value: 65535

cip

Enable or disable the insertion of the actual client IP address into the HTTP header request passed from the client to one, some, or all servers attached to the system. The passed address can then be accessed through a minor modification to the server.

* If the CIP header is specified, it will be used as the client IP header.

* If the CIP header is not specified, the value that has been set will be used as the client IP header.

Possible values: ENABLED, DISABLED

cipHeader

Text that will be used as the client IP address header.

cookieversion

Version of the cookie inserted by the system.

Possible values: 0, 1

secureCookie

Enable or disable secure flag for persistence cookie.

Possible values: ENABLED, DISABLED

Default value: ENABLED

pmtuMin

Minimum path MTU value that NetScaler will process in the ICMP fragmentation needed message. If the ICMP message contains a value less than this value, then this value is used instead.

Default value: 576

Minimum value: 168

Maximum value: 1500

pmtuTimeout

Interval, in minutes, for flushing the PMTU entries.

Default value: 10

Minimum value: 1

Maximum value: 1440

ftpPortRange

Minimum and maximum port (port range) that FTP services are allowed to use.

Minimum value: 1024

Maximum value: 64000

crPortRange

Port range for cache redirection services.

Minimum value: 1

Maximum value: 65535

timezone

Time zone for the NetScaler appliance. Name of the time zone should be specified as argument.

Possible values: CoordinatedUniversalTime, GMT+01:00-CET-Europe/Andorra, GMT+04:00-GST-Asia/Dubai, GMT+04:30-AFT-Asia/Kabul, GMT-04:00-AST-America/Antigua, GMT-04:00-AST-America/Anguilla, GMT+01:00-CET-Europe/Tirane, GMT+04:00-AMT-Asia/Yerevan, GMT+01:00-WAT-Africa/Luanda, GMT+13:00-NZDT-Antarctica/McMurdo, GMT+13:00-NZDT-Antarctica/South_Pole, GMT-03:00-ROTT-Antarctica/Rothera, GMT-04:00-CLT-Antarctica/Palmer, GMT+05:00-MAWT-Antarctica/Mawson, GMT+07:00-DAVT-Antarctica/Davis, GMT+08:00-WST-Antarctica/Casey, GMT+06:00-VOST-Antarctica/Vostok, GMT+10:00-DDUT-Antarctica/DumontDUrville, GMT+03:00-SYOT-Antarctica/Syowa, GMT+11:00-MIST-Antarctica/Macquarie, GMT-03:00-ART-America/Argentina/Buenos_Aires, GMT-03:00-ART-America/Argentina/Cordoba, GMT-03:00-ART-America/Argentina/Salta, GMT-03:00-ART-America/Argentina/Jujuy, GMT-03:00-ART-America/Argentina/Tucuman, GMT-03:00-ART-America/Argentina/Catamarca, GMT-03:00-ART-America/Argentina/La_Rioja, GMT-03:00-ART-America/Argentina/San_Juan, GMT-03:00-ART-America/Argentina/Mendoza, GMT-03:00-WARST-America/Argentina/San_Luis, GMT-03:00-ART-America/Argentina/Rio_Gallegos, GMT-03:00-ART-America/Argentina/Ushuaia, GMT-11:00-SST-Pacific/Pago_Pago, GMT+01:00-CET-Europe/Vienna, GMT+11:00-LHST-Australia/Lord_Howe, GMT+11:00-EST-Australia/Hobart, GMT+11:00-EST-Australia/Currie, GMT+11:00-EST-Australia/Melbourne, GMT+11:00-EST-Australia/Sydney, GMT+10:30-

CST-Australia/Broken_Hill, GMT+10:00-EST-Australia/Brisbane, GMT+10:00-EST-Australia/Lindeman, GMT+10:30-CST-Australia/Adelaide, GMT+09:30-CST-Australia/Darwin, GMT+08:00-WST-Australia/Perth, GMT+08:45-CWST-Australia/Eucla, GMT-04:00-AST-America/Aruba, GMT+02:00-EET-Europe/Mariehamn, GMT+04:00-AZT-Asia/Baku, GMT+01:00-CET-Europe/Sarajevo, GMT-04:00-AST-America/Barbados, GMT+06:00-BDT-Asia/Dhaka, GMT+01:00-CET-Europe/Brussels, GMT+00:00-GMT-Africa/Ouagadougou, GMT+02:00-EET-Europe/Sofia, GMT+03:00-AST-Asia/Bahrain, GMT+02:00-CAT-Africa/Bujumbura, GMT+01:00-WAT-Africa/Porto-Novo, GMT-04:00-AST-America/St_Barthelemy, GMT-03:00-ADT-Atlantic/Bermuda, GMT+08:00-BNT-Asia/Brunei, GMT-04:00-BOT-America/La_Paz, GMT-02:00-FNT-America/Noronha, GMT-03:00-BRT-America/Belem, GMT-03:00-BRT-America/Fortaleza, GMT-03:00-BRT-America/Recife, GMT-03:00-BRT-America/Araguaina, GMT-03:00-BRT-America/Maceio, GMT-03:00-BRT-America/Bahia, GMT-03:00-BRT-America/Sao_Paulo, GMT-04:00-AMT-America/Campo_Grande, GMT-04:00-AMT-America/Cuiaba, GMT-03:00-BRT-America/Santarem, GMT-04:00-AMT-America/Porto_Velho, GMT-04:00-AMT-America/Boa_Vista, GMT-04:00-AMT-America/Manaus, GMT-04:00-AMT-America/Eirunepe, GMT-04:00-AMT-America/Rio_Branco, GMT-04:00-EDT-America/Nassau, GMT+06:00-BTT-Asia/Thimphu, GMT+02:00-CAT-Africa/Gaborone, GMT+03:00-FET-Europe/Minsk, GMT-06:00-CST-America/Belize, GMT-02:30-NDT-America/St_Johns, GMT-03:00-ADT-America/Halifax, GMT-03:00-ADT-America/Glace_Bay, GMT-03:00-ADT-America/Moncton, GMT-03:00-ADT-America/Goose_Bay, GMT-04:00-AST-America/Blanc-Sablon, GMT-04:00-EDT-America/Montreal, GMT-04:00-EDT-America/Toronto, GMT-04:00-EDT-America/Nipigon, GMT-04:00-EDT-America/Thunder_Bay, GMT-04:00-EDT-America/Iqaluit, GMT-04:00-EDT-America/Pangnirtung, GMT-05:00-CDT-America/Resolute, GMT-05:00-EST-America/Atikokan, GMT-05:00-CDT-America/Rankin_Inlet, GMT-05:00-CDT-America/Winnipeg, GMT-05:00-CDT-America/Rainy_River, GMT-06:00-CST-America/Regina, GMT-06:00-CST-America/Swift_Current, GMT-06:00-MDT-America/Edmonton, GMT-06:00-MDT-America/Cambridge_Bay, GMT-06:00-MDT-America/Yellowknife, GMT-06:00-MDT-America/Inuvik, GMT-07:00-MST-America/Dawson_Creek, GMT-07:00-PDT-America/Vancouver, GMT-07:00-PDT-America/Whitehorse, GMT-07:00-PDT-America/Dawson, GMT+06:30-CCT-Indian/Cocos, GMT+01:00-WAT-Africa/Kinshasa, GMT+02:00-CAT-Africa/Lubumbashi, GMT+01:00-WAT-Africa/Bangui, GMT+01:00-WAT-Africa/Brazzaville, GMT+01:00-CET-Europe/Zurich, GMT+00:00-GMT-Africa/Abidjan, GMT-10:00-CKT-Pacific/Rarotonga, GMT-04:00-CLT-America/Santiago, GMT-06:00-EAST-Pacific/Easter, GMT+01:00-WAT-Africa/Douala, GMT+08:00-CST-Asia/Shanghai, GMT+08:00-CST-Asia/Harbin, GMT+08:00-CST-Asia/Chongqing, GMT+08:00-CST-Asia/Urumqi, GMT+08:00-CST-Asia/Kashgar, GMT-05:00-COT-America/Bogota, GMT-06:00-CST-America/Costa_Rica, GMT-04:00-CDT-America/Havana, GMT-01:00-CVT-Atlantic/Cape_Verde, GMT+07:00-CXT-Indian/Christmas, GMT+02:00-EET-Asia/Nicosia, GMT+01:00-CET-Europe/Prague, GMT+01:00-CET-Europe/Berlin, GMT+03:00-EAT-Africa/Djibouti, GMT+01:00-CET-Europe/Copenhagen, GMT-04:00-AST-America/Dominica, GMT-04:00-AST-America/Santo_Domingo, GMT+01:00-CET-Africa/Algiers, GMT-05:00-ECT-America/Guayaquil, GMT-06:00-GALT-Pacific/Galapagos, GMT+02:00-EET-Europe/Tallinn, GMT+02:00-EET-Africa/Cairo, GMT+00:00-WET-Africa/El_Aaiun, GMT+03:00-EAT-Africa/Asmara, GMT+01:00-CET-Europe/Madrid, GMT+01:00-CET-Africa/Ceuta, GMT+00:00-WET-Atlantic/Canary, GMT+03:00-EAT-Africa/Addis_Ababa, GMT+02:00-EET-Europe/Helsinki, GMT+12:00-FJT-Pacific/Fiji, GMT-03:00-FKST-Atlantic/Stanley, GMT+10:00-CHUT-Pacific/Chuuk, GMT+11:00-PONT-Pacific/Pohnpei, GMT+11:00-KOST-Pacific/Kosrae, GMT+00:00-WET-Atlantic/Faroe, GMT+01:00-CET-Europe/Paris, GMT+01:00-WAT-Africa/Libreville, GMT+00:00-GMT-Europe/London, GMT-04:00-AST-America/Grenada, GMT+04:00-GET-Asia/Tbilisi, GMT-03:00-GFT-America/Cayenne, GMT+00:00-GMT-Europe/Guernsey, GMT+00:00-GMT-Africa/Accra, GMT+01:00-CET-Europe/Gibraltar, GMT-03:00-WGT-America/Godthab, GMT+00:00-GMT-America/Danmarkshavn, GMT-01:00-EGT-America/Scoresbysund, GMT-03:00-ADT-America/Thule, GMT+00:00-GMT-Africa/Banjul, GMT+00:00-GMT-Africa/Conakry, GMT-04:00-AST-America/Guadeloupe, GMT+01:00-WAT-Africa/Malabo, GMT+02:00-EET-Europe/Athens, GMT-02:00-GST-Atlantic/South_Georgia, GMT-06:00-CST-America/Guatemala, GMT+10:00-ChST-Pacific/Guam, GMT+00:00-GMT-Africa/Bissau, GMT-04:00-GYT-America/Guyana, GMT+08:00-HKT-Asia/Hong_Kong, GMT-06:00-CST-America/Tegucigalpa, GMT+01:00-CET-Europe/Zagreb, GMT-05:00-EST-America/Port-au-Prince, GMT+01:00-CET-Europe/Budapest, GMT+07:00-WIT-Asia/Jakarta, GMT+07:00-WIT-Asia/Pontianak, GMT+08:00-CIT-Asia/Makassar, GMT+09:00-EIT-Asia/Jayapura, GMT+00:00-GMT-Europe/Dublin, GMT+02:00-IST-Asia/Jerusalem, GMT+00:00-GMT-Europe/Isle_of_Man, GMT+05:30-IST-Asia/Kolkata, GMT+06:00-IOT-Indian/Chagos, GMT+03:00-AST-Asia/Baghdad, GMT+03:30-IRST-Asia/Tehran, GMT+00:00-GMT-Atlantic/Reykjavik, GMT+01:00-CET-Europe/Rome, GMT+00:00-GMT-Europe/Jersey, GMT-05:00-EST-America/Jamaica, GMT+02:00-EET-Asia/Amman, GMT+09:00-JST-Asia/Tokyo, GMT+03:00-EAT-Africa/Nairobi, GMT+06:00-KGT-Asia/Bishkek, GMT+07:00-ICT-Asia/Phnom_Penh, GMT+12:00-GILT-Pacific/Tarawa, GMT+13:00-PHOT-Pacific/Enderbury, GMT+14:00-LINT-Pacific/Kiritimati, GMT+03:00-EAT-Indian/Comoro, GMT-04:00-AST-America/St_Kitts, GMT+09:00-KST-Asia/Pyongyang, GMT+09:00-KST-Asia/Seoul, GMT+03:00-AST-Asia/Kuwait, GMT-05:00-EST-America/Cayman, GMT+06:00-ALMT-Asia/Almaty, GMT+06:00-QYZT-Asia/Qyzylorda, GMT+05:00-AQTT-Asia/Aqtobe, GMT+05:00-AQTT-Asia/Aqtau, GMT+05:00-ORAT-Asia/Oral, GMT+07:00-ICT-Asia/Vientiane, GMT+02:00-EET-Asia/Beirut, GMT-04:00-AST-America/St_Lucia, GMT+01:00-CET-Europe/Vaduz, GMT+05:30-IST-Asia/Colombo, GMT+00:00-GMT-Africa/Monrovia, GMT+02:00-SAST-Africa/Maseru, GMT+02:00-EET-Europe/Vilnius, GMT+01:00-CET-Europe/Luxembourg, GMT+02:00-EET-Europe/Riga, GMT+02:00-EET-Africa/Tripoli, GMT+00:00-WET-Africa/Casablanca, GMT+01:00-CET-Europe/Monaco, GMT+02:00-EET-Europe/Chisinau, GMT+01:00-CET-Europe/Podgorica, GMT-04:00-AST-America/Marigot, GMT+03:00-EAT-Indian/Antananarivo, GMT+12:00-MHT-Pacific/Majuro, GMT+12:00-MHT-Pacific/Kwajalein, GMT+01:00-CET-Europe/Skopje, GMT+00:00-GMT-Africa/Bamako, GMT+06:30-MMT-Asia/Rangoon, GMT+08:00-ULAT-Asia/Ulaanbaatar, GMT+07:00-HOVT-Asia/Hovd, GMT+08:00-CHOT-Asia/Choibalsan, GMT+08:00-CST-Asia/Macau, GMT+10:00-ChST-Pacific/Saipan, GMT-04:00-AST-America/Martinique, GMT+00:00-GMT-Africa/Nouakchott, GMT-04:00-AST-America/Montserrat,

GMT+01:00-CET-Europe/Malta, GMT+04:00-MUT-Indian/Mauritius, GMT+05:00-MVT-Indian/Maldives, GMT+02:00-CAT-Africa/Blantyre, GMT-06:00-CST-America/Mexico_City, GMT-06:00-CST-America/Cancun, GMT-06:00-CST-America/Merida, GMT-06:00-CST-America/Monterrey, GMT-05:00-CDT-America/Matamoros, GMT-07:00-MST-America/Mazatlan, GMT-07:00-MST-America/Chihuahua, GMT-06:00-MDT-America/Ojinaga, GMT-07:00-MST-America/Hermosillo, GMT-07:00-PDT-America/Tijuana, GMT-08:00-PST-America/Santa_Isabel, GMT-06:00-CST-America/Bahia_Banderas, GMT+08:00-MYT-Asia/Kuala_Lumpur, GMT+08:00-MYT-Asia/Kuching, GMT+02:00-CAT-Africa/Maputo, GMT+02:00-WAST-Africa/Windhoek, GMT+11:00-NCT-Pacific/Noumea, GMT+01:00-WAT-Africa/Niamey, GMT+11:30-NFT-Pacific/Norfolk, GMT+01:00-WAT-Africa/Lagos, GMT-06:00-CST-America/Managua, GMT+01:00-CET-Europe/Amsterdam, GMT+01:00-CET-Europe/Oslo, GMT+05:45-NPT-Asia/Kathmandu, GMT+12:00-NRT-Pacific/Nauru, GMT-11:00-NUT-Pacific/Niue, GMT+13:00-NZDT-Pacific/Auckland, GMT+13:45-CHADT-Pacific/Chatham, GMT+04:00-GST-Asia/Muscat, GMT-05:00-EST-America/Panama, GMT-05:00-PET-America/Lima, GMT-10:00-TAHT-Pacific/Tahiti, GMT-09:30-MART-Pacific/Marquesas, GMT-09:00-GAMT-Pacific/Gambier, GMT+10:00-PGT-Pacific/Port_Moresby, GMT+08:00-PHT-Asia/Manila, GMT+05:00-PKT-Asia/Karachi, GMT+01:00-CET-Europe/Warsaw, GMT-02:00-PMDT-America/Miquelon, GMT-08:00-PST-Pacific/Pitcairn, GMT-04:00-AST-America/Puerto_Rico, GMT+02:00-EET-Asia/Gaza, GMT+02:00-EET-Asia/Hebron, GMT+00:00-WET-Europe/Lisbon, GMT+00:00-WET-Atlantic/Madeira, GMT-01:00-AZOT-Atlantic/Azores, GMT+09:00-PWT-Pacific/Palau, GMT-03:00-PYST-America/Asuncion, GMT+03:00-AST-Asia/Qatar, GMT+04:00-RET-Indian/Reunion, GMT+02:00-EET-Europe/Bucharest, GMT+01:00-CET-Europe/Belgrade, GMT+03:00-FET-Europe/Kaliningrad, GMT+04:00-MSK-Europe/Moscow, GMT+04:00-VOLT-Europe/Volgograd, GMT+04:00-SAMT-Europe/Samara, GMT+06:00-YEKT-Asia/Yekaterinburg, GMT+07:00-OMST-Asia/Omsk, GMT+07:00-NOVT-Asia/Novosibirsk, GMT+07:00-NOVT-Asia/Novokuznetsk, GMT+08:00-KRAT-Asia/Krasnoyarsk, GMT+09:00-IRKT-Asia/Irkutsk, GMT+10:00-YAKT-Asia/Yakutsk, GMT+11:00-VLAT-Asia/Vladivostok, GMT+11:00-SAKT-Asia/Sakhalin, GMT+12:00-MAGT-Asia/Magadan, GMT+12:00-PETT-Asia/Kamchatka, GMT+12:00-ANAT-Asia/Anadyr, GMT+02:00-CAT-Africa/Kigali, GMT+03:00-AST-Asia/Riyadh, GMT+11:00-SBT-Pacific/Guadacanal, GMT+04:00-SCT-Indian/Mahe, GMT+03:00-EAT-Africa/Khartoum, GMT+01:00-CET-Europe/Stockholm, GMT+08:00-SGT-Asia/Singapore, GMT+00:00-GMT-Atlantic/St_Helena, GMT+01:00-CET-Europe/Ljubljana, GMT+01:00-CET-Arctic/Longyearbyen, GMT+01:00-CET-Europe/Bratislava, GMT+00:00-GMT-Africa/Freetown, GMT+01:00-CET-Europe/San_Marino, GMT+00:00-GMT-Africa/Dakar, GMT+03:00-EAT-Africa/Mogadishu, GMT-03:00-SRT-America/Paramaribo, GMT+00:00-GMT-Africa/Sao_Tome, GMT-06:00-CST-America/El_Salvador, GMT+02:00-EET-Asia/Damascus, GMT+02:00-SAST-Africa/Mbabane, GMT-04:00-EDT-America/Grand_Turk, GMT+01:00-WAT-Africa/Ndjamena, GMT+05:00-TFT-Indian/Kerguelen, GMT+00:00-GMT-Africa/Lome, GMT+07:00-ICT-Asia/Bangkok, GMT+05:00-TJT-Asia/Dushanbe, GMT-10:00-TKT-Pacific/Fakaofu, GMT+09:00-TLT-Asia/Dili, GMT+05:00-TMT-Asia/Ashgabat, GMT+01:00-CET-Africa/Tunis, GMT+13:00-TOT-Pacific/Tongatapu, GMT+02:00-EET-Europe/Istanbul, GMT-04:00-AST-America/Port_of_Spain, GMT+12:00-TVT-Pacific/Funafuti, GMT+08:00-CST-Asia/Taipei, GMT+03:00-EAT-Africa/Dar_es_Salaam, GMT+02:00-EET-Europe/Kiev, GMT+02:00-EET-Europe/Uzhgorod, GMT+02:00-EET-Europe/Zaporozhye, GMT+02:00-EET-Europe/Simferopol, GMT+03:00-EAT-Africa/Kampala, GMT-10:00-HST-Pacific/Johnston, GMT-11:00-SST-Pacific/Midway, GMT+12:00-WAKT-Pacific/Wake, GMT-04:00-EDT-America/New_York, GMT-04:00-EDT-America/Detroit, GMT-04:00-EDT-America/Kentucky/Louisville, GMT-04:00-EDT-America/Kentucky/Monticello, GMT-04:00-EDT-America/Indiana/Indianapolis, GMT-04:00-EDT-America/Indiana/Vincennes, GMT-04:00-EDT-America/Indiana/Winamac, GMT-04:00-EDT-America/Indiana/Marengo, GMT-04:00-EDT-America/Indiana/Petersburg, GMT-04:00-EDT-America/Indiana/Vevay, GMT-05:00-CDT-America/Chicago, GMT-05:00-CDT-America/Indiana/Tell_City, GMT-05:00-CDT-America/Indiana/Knox, GMT-05:00-CDT-America/Menominee, GMT-05:00-CDT-America/North_Dakota/Center, GMT-05:00-CDT-America/North_Dakota/New_Salem, GMT-05:00-CDT-America/North_Dakota/Beulah, GMT-06:00-MDT-America/Denver, GMT-06:00-MDT-America/Boise, GMT-06:00-MDT-America/Shiprock, GMT-07:00-MST-America/Phoenix, GMT-07:00-PDT-America/Los_Angeles, GMT-08:00-AKDT-America/Anchorage, GMT-08:00-AKDT-America/Juneau, GMT-08:00-AKDT-America/Sitka, GMT-08:00-AKDT-America/Yakutat, GMT-08:00-AKDT-America/Nome, GMT-09:00-HADT-America/Adak, GMT-08:00-MeST-America/Metlakatla, GMT-10:00-HST-Pacific/Honolulu, GMT-03:00-UYT-America/Montevideo, GMT+05:00-UZT-Asia/Samarkand, GMT+05:00-UZT-Asia/Tashkent, GMT+01:00-CET-Europe/Vatican, GMT-04:00-AST-America/St_Vincent, GMT-04:30-VET-America/Caracas, GMT-04:00-AST-America/Tortola, GMT-04:00-AST-America/St_Thomas, GMT+07:00-ICT-Asia/Ho_Chi_Minh, GMT+11:00-VUT-Pacific/Efate, GMT+12:00-WFT-Pacific/Wallis, GMT+14:00-WSDT-Pacific/Apia, GMT+03:00-AST-Asia/Aden, GMT+03:00-EAT-Indian/Mayotte, GMT+02:00-SAST-Africa/Johannesburg, GMT+02:00-CAT-Africa/Lusaka, GMT+02:00-CAT-Africa/Harare

grantQuotaMaxClient

Percentage of shared quota to be granted at a time for maxClient.

Default value: 10

Minimum value: 0

Maximum value: 100

exclusiveQuotaMaxClient

Percentage of maxClient to be given to PEs.

Default value: 80

Minimum value: 0

Maximum value: 100

grantQuotaSpillOver

Percentage of shared quota to be granted at a time for spillover.

Default value: 10

Minimum value: 0

Maximum value: 100

exclusiveQuotaSpillOver

Percentage of maximum limit to be given to PEs.

Default value: 80

Minimum value: 0

Maximum value: 100

useproxyport

Enable/Disable use_proxy_port setting

Possible values: ENABLED, DISABLED

Default value: ENABLED

internaluserlogin

Enables/disables the internal user from logging in to the appliance. Before disabling internal user login, you must have key-based authentication set up on the appliance. The file name for the key pair must be "ns_comm_key".

Possible values: ENABLED, DISABLED

Default value: ENABLED

aftpAllowRandomSourcePort

Allow the FTP server to come from a random source port for active FTP data connections

Possible values: ENABLED, DISABLED

Default value: DISABLED

icaPorts

The ICA ports on the Web server. This allows the system to perform connection off-load for any client request that has a destination port matching one of these configured ports.

Minimum value: 1

tcpCIP

Enable or disable the insertion of the client TCP/IP header in TCP payload passed from the client to one, some, or all servers attached to the system. The passed address can then be accessed through a minor modification to the server.

Possible values: ENABLED, DISABLED

Default value: DISABLED

unset ns param

Removes the attributes of the NetScaler parameters. Attributes for which a default value is available revert to their default values. Refer to the 'set ns param' command for a description of the parameters..Refer to the set ns param command for meanings of the arguments.

Synopsys

```
unset ns param [-ftpPortRange] [-crPortRange] [-timezone] [-aftpAllowRandomSourcePort] [-httpPort] [-maxConn] [-maxReq] [-cip] [-cipHeader] [-cookieversion] [-secureCookie] [-pmtuMin] [-pmtuTimeout] [-grantQuotaMaxClient] [-exclusiveQuotaMaxClient] [-grantQuotaSpillOver] [-exclusiveQuotaSpillOver] [-useproxyport] [-internaluserlogin] [-icaPorts] [-tcpCIP]
```

show ns param

Displays the information of the parameters of the NetScaler appliance that were set by using the 'set ns param' command.

Synopsys

```
show ns param
```

Outputs

httpPort

The HTTP ports on the Web server.

maxConn

Maximum Number of Connections.

maxReq

Maximum Number of requests that can be handled.

cip

Insertion of client IP address into the HTTP header.

cipHeader

The text that will be used as the client IP header.

cookieversion

Version of the cookie inserted by the system.

secureCookie

Enable or disable secure flag for persistence cookie.

pmtuMin

Minimum path MTU value that NetScaler will process in the ICMP fragmentation needed message. If the ICMP message contains a value less than this value, then this value is used instead.

pmtuTimeout

Interval, in minutes, for flushing the PMTU entries.

ftpPortRange

Minimum and maximum port (port range) that FTP services are allowed to use.

crPortRange

Port range for cache redirection services.

timezone

Time zone for the NetScaler appliance. Name of the time zone should be specified as argument.

grantQuotaMaxClient

Percentage of shared quota to be granted at a time for maxClient.

exclusiveQuotaMaxClient

Percentage of maxClient to be given to PEs.

grantQuotaSpillOver

Percentage of shared quota to be granted at a time for spillover.

exclusiveQuotaSpillOver

Percentage of maximum limit to be given to PEs.

useproxyport

Enable/Disable use_proxy_port setting

internaluserlogin

Enables/disables the internal user from logging in to the appliance. Before disabling internal user login, you must have key-based authentication set up on the appliance. The file name for the key pair must be "ns_comm_key".

aftpAllowRandomSourcePort

Allow the FTP server to come from a random source port for active FTP data connections

icaPorts

The ICA ports on the Web server. This allows the system to perform connection off-load for any client request that has a destination port matching one of these configured ports.

tcpCIP

Enable or disable the insertion of the client TCP/IP header in TCP payload passed from the client to one, some, or all servers attached to the system. The passed address can then be accessed through a minor modification to the server.

ns pbr

The following operations can be performed on "ns pbr":

add | **rm** | **set** | **unset** | **enable** | **disable** | **stat** | **show**

add ns pbr

Adds a policy based route (PBR) to the NetScaler appliance. To commit this operation, you must apply the PBRs. A PBR specifies criteria for selecting outgoing IPv4 packets and, typically, a next hop to which to send the selected packets. For example, you can configure the NetScaler appliance to route outgoing packets from a specific IP address or range to a particular next hop router. Note: The NetScaler appliance process PBRs before processing the RNAT rules.

Synopsys

```
add ns pbr <name> <action> [-td <positive_integer>] [-srcIP [<operator>] <srcIPVal>] [-srcPort [<operator>] <srcPortVal>] [-destIP [<operator>] <destIPVal>] [-destPort [<operator>] <destPortVal>] ((-nextHop <nextHopVal>) | (-ipTunnel <ipTunnelName>)) [-srcMac <mac_addr>] [-protocol <protocol> | -protocolNumber <positive_integer>] [-vlan <positive_integer> | -vxlan <positive_integer>] [-interface <interface_name>] [-priority <positive_integer>] [-msr ( ENABLED | DISABLED )] [-monitor <string>]] [-state ( ENABLED | DISABLED )]
```

Arguments

name

Name for the PBR. Must begin with an ASCII alphabetic or underscore `\\(_\\)` character, and must contain only ASCII alphanumeric, underscore, hash `\\(\\#\\)`, period `\\(\\.\\)`, space, colon `\\(:\\)`, at `\\(@\\)`, equals `\\(=\\)`, and hyphen `\\(-\\)` characters. Can be changed after the PBR is created.

action

Action to perform on the outgoing IPv4 packets that match the PBR.

Available settings function as follows:

- * ALLOW - The NetScaler appliance sends the packet to the designated next-hop router.
- * DENY - The NetScaler appliance applies the routing table for normal destination-based routing.

Possible values: ALLOW, DENY

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

srcIP

IP address or range of IP addresses to match against the source IP address of an outgoing IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [10.102.29.30-10.102.29.189].

operator

Logical operator.

Possible values: =, !=, EQ, NEQ

srcIPVal

IP address or range of IP addresses to match against the source IP address of an outgoing IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [10.102.29.30-10.102.29.189].

srcPort

Port number or range of port numbers to match against the source port number of an outgoing IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

Note: The destination port can be specified only for TCP and UDP protocols.

srcPortVal

Port number or range of port numbers to match against the source port number of an outgoing IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

Note: The destination port can be specified only for TCP and UDP protocols.

Maximum value: 65535

destIP

IP address or range of IP addresses to match against the destination IP address of an outgoing IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [10.102.29.30-10.102.29.189].

destIPVal

IP address or range of IP addresses to match against the destination IP address of an outgoing IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [10.102.29.30-10.102.29.189].

destPort

Port number or range of port numbers to match against the destination port number of an outgoing IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

Note: The destination port can be specified only for TCP and UDP protocols.

destPortVal

Port number or range of port numbers to match against the destination port number of an outgoing IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

Note: The destination port can be specified only for TCP and UDP protocols.

Maximum value: 65535

nextHop

IP address of the next hop router or the name of the link load balancing virtual server to which to send matching packets if action is set to ALLOW.

If you specify a link load balancing (LLB) virtual server, which can provide a backup if a next hop link fails, first make sure that the next hops bound to the LLB virtual server are actually next hops that are directly connected to the NetScaler appliance. Otherwise, the NetScaler throws an error when you attempt to create the PBR.

nextHopVal

The Next Hop IP address or gateway name.

ipTunnel

The Tunnel name.

ipTunnelName

The iptunnel name where packets need to be forwarded upon.

srcMac

MAC address to match against the source MAC address of an outgoing IPv4 packet.

protocol

Protocol, identified by protocol name, to match against the protocol of an outgoing IPv4 packet.

Possible values: ICMP, IGMP, TCP, EGP, IGP, ARGUS, UDP, RDP, RSVP, EIGRP, L2TP, ISIS

protocolNumber

Protocol, identified by protocol number, to match against the protocol of an outgoing IPv4 packet.

Minimum value: 1

Maximum value: 255

vlan

ID of the VLAN. The NetScaler appliance compares the PBR only to the outgoing packets on the specified VLAN. If you do not specify any interface ID, the appliance compares the PBR to the outgoing packets on all VLANs.

Minimum value: 1

Maximum value: 4094

vxlan

ID of the VXLAN. The NetScaler appliance compares the PBR only to the outgoing packets on the specified VXLAN. If you do not specify any interface ID, the appliance compares the PBR to the outgoing packets on all VXLANs.

Minimum value: 1

Maximum value: 16777215

interface

ID of an interface. The NetScaler appliance compares the PBR only to the outgoing packets on the specified interface. If you do not specify any value, the appliance compares the PBR to the outgoing packets on all interfaces.

priority

Priority of the PBR, which determines the order in which it is evaluated relative to the other PBRs. If you do not specify priorities while creating PBRs, the PBRs are evaluated in the order in which they are created.

Minimum value: 1

Maximum value: 81920

msr

Monitor the route specified by Next Hop parameter. This parameter is not applicable if you specify a link load balancing (LLB) virtual server name with the Next Hop parameter.

Possible values: ENABLED, DISABLED

Default value: DISABLED

monitor

The name of the monitor. (Can be only of type ping or ARP)

state

Enable or disable the PBR. After you apply the PBRs, the NetScaler appliance compares outgoing packets to the enabled PBRs.

Possible values: ENABLED, DISABLED

Default value: ENABLED

Example

```
add ns pbr a allow -srcip 10.102.37.252 -destip 10.10.10.2 -nexthop 11.11.11.2
```

rm ns pbr

Removes a PBR from the NetScaler appliance. To commit this operation, you must apply the PBRs.

Synopsys

```
rm ns pbr <name> ...
```

Arguments

name

Name of the PBR that you want to remove.

Example

```
rm ns pbr a
```

set ns pbr

Modifies the specified parameters of a PBR. To commit this operation, you must apply the PBRs.

Synopsys

```
set ns pbr <name> [-action ( ALLOW | DENY )] [-srcIP [<operator>] <srcIPVal>] [-srcPort [<operator>] <srcPortVal>] [-destIP [<operator>] <destIPVal>] [-destPort [<operator>] <destPortVal>] ((-nextHop <nextHopVal>) | (-ipTunnel <ipTunnelName>)) [-srcMac <mac_addr>] [-protocol <protocol> | -protocolNumber <positive_integer>] [-vlan <positive_integer> | -vxlan <positive_integer>] [-interface <interface_name>] [-priority <positive_integer>] [-msr ( ENABLED | DISABLED )] [-monitor <string>]]
```

Arguments

name

Name of the PBR whose parameters you want to modify.

action

Action to perform on the outgoing IPv4 packets that match the PBR.

Available settings function as follows:

* ALLOW - The NetScaler appliance sends the packet to the designated next-hop router.

* DENY - The NetScaler appliance applies the routing table for normal destination-based routing.

Possible values: ALLOW, DENY

srcIP

IP address or range of IP addresses to match against the source IP address of an outgoing IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [10.102.29.30-10.102.29.189].

operator

Logical operator.

Possible values: =, !=, EQ, NEQ

srcIPVal

IP address or range of IP addresses to match against the source IP address of an outgoing IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [10.102.29.30-10.102.29.189].

srcPort

Port number or range of port numbers to match against the source port number of an outgoing IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

Note: The destination port can be specified only for TCP and UDP protocols.

srcPortVal

Port number or range of port numbers to match against the source port number of an outgoing IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

Note: The destination port can be specified only for TCP and UDP protocols.

Maximum value: 65535

destIP

IP address or range of IP addresses to match against the destination IP address of an outgoing IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [10.102.29.30-10.102.29.189].

destIPVal

IP address or range of IP addresses to match against the destination IP address of an outgoing IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [10.102.29.30-10.102.29.189].

destPort

Port number or range of port numbers to match against the destination port number of an outgoing IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

Note: The destination port can be specified only for TCP and UDP protocols.

destPortVal

Port number or range of port numbers to match against the destination port number of an outgoing IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

Note: The destination port can be specified only for TCP and UDP protocols.

Maximum value: 65535

nextHop

IP address of the next hop router or the name of the link load balancing virtual server to which to send matching packets if action is set to ALLOW.

If you specify a link load balancing (LLB) virtual server, which can provide a backup if a next hop link fails, first make sure that the next hops bound to the LLB virtual server are actually next hops that are directly connected to the NetScaler appliance. Otherwise, the NetScaler throws an error when you attempt to create the PBR.

nextHopVal

The Next Hop IP address or gateway name.

ipTunnel

The Tunnel name.

ipTunnelName

The iptunnel name where packets need to be forwarded upon.

srcMac

MAC address to match against the source MAC address of an outgoing IPv4 packet.

protocol

Protocol, identified by protocol name, to match against the protocol of an outgoing IPv4 packet.

Possible values: ICMP, IGMP, TCP, EGP, IGP, ARGUS, UDP, RDP, RSVP, EIGRP, L2TP, ISIS

protocolNumber

Protocol, identified by protocol number, to match against the protocol of an outgoing IPv4 packet.

Minimum value: 1

Maximum value: 255

vlan

ID of the VLAN. The NetScaler appliance compares the PBR only to the outgoing packets on the specified VLAN. If you do not specify any interface ID, the appliance compares the PBR to the outgoing packets on all VLANs.

Minimum value: 1

Maximum value: 4094

vxlan

ID of the VXLAN. The NetScaler appliance compares the PBR only to the outgoing packets on the specified VXLAN. If you do not specify any interface ID, the appliance compares the PBR to the outgoing packets on all VXLANs.

Minimum value: 1

Maximum value: 16777215

interface

ID of an interface. The NetScaler appliance compares the PBR only to the outgoing packets on the specified interface. If you do not specify any value, the appliance compares the PBR to the outgoing packets on all interfaces.

priority

Priority of the PBR, which determines the order in which it is evaluated relative to the other PBRs. If you do not specify priorities while creating PBRs, the PBRs are evaluated in the order in which they are created.

Minimum value: 1

Maximum value: 81920

msr

Monitor the route specified by the Next Hop parameter. This parameter is not applicable if you specify a link load balancing (LLB) virtual server name with the Next Hop parameter.

Possible values: ENABLED, DISABLED

Default value: DISABLED

monitor

Name of the monitor, of type PING or ARP, configured on the NetScaler appliance to monitor the route specified by the Next Hop parameter. You must enable the MSR parameter before setting this parameter.

Example

```
set ns pbr a -srcPort 50
```

unset ns pbr

Resets the attributes of the specified PBR. Attributes for which a default value is available revert to their default values. Refer to the set ns pbr command for descriptions of the parameters..Refer to the set ns pbr command for meanings of the arguments.

Synopsys

```
unset ns pbr <name> [-srcIP] [-srcPort] [-destIP] [-destPort] [-nextHop] [-ipTunnel] [-srcMac] [-protocol] [-vlan] [-vxlan] [-interface] [-msr] [-monitor]
```

Example

```
unset ns pbr rule1 -srcPort
```

enable ns pbr

Enables a PBR. To commit this operation, you must apply the PBRs. After you apply the PBRs, the NetScaler appliance compares outgoing packets to the enabled PBRs.

Synopsys

```
enable ns pbr <name> ...
```

Arguments

name

Name of PBR that you want to enable.

Example

```
enable ns pbr foo
```

disable ns pbr

Disables a PBR. To commit this operation, you must apply the PBRs. After you apply the PBRs, the NetScaler appliance does not compare outgoing packets against the disabled PBRs

Synopsys

```
disable ns pbr <name> ...
```

Arguments

name

Name of PBR that you want to disable.

Example

```
disable ns pbr foo
```

stat ns pbr

Displays statistics related to the PBRs. To display statistics of all the PBRs, run the command without any parameters. To display statistics of a particular PBR, specify the name of the PBR.

Synopsys

```
stat ns pbr [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

name

Name of the PBR whose statistics you want the NetScaler appliance to display.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Allow PBR hits (PBRAllow)

Total packets that matched the PBR (Policy-Based Routes) with action ALLOW

Deny PBR hits (PBRDeny)

Total packets that matched the PBR with action DENY

PBR hits (PBRTotHits)

Total packets that matched one of the configured PBR

PBR misses (PBRMiss)

Total packets that did not match any PBR

Hits for this PBR (PBRHits)

Number of times the pbr was hit

Example

```
stat pbr
```

show ns pbr

Displays settings related to the PBRs. To display settings of all the PBRs, run the command without any parameters. To display settings of a particular PBR, specify the name of the PBR.

Synopsys

```
show ns pbr [<name>] [-detail]
```

Arguments

name

Name of the PBR whose details you want the NetScaler appliance to display.

detail

To get a detailed view.

Outputs

action

Action to perform on the outgoing IPv4 packets that match the PBR.

Available settings function as follows:

- * ALLOW - The NetScaler appliance sends the packet to the designated next-hop router.
- * DENY - The NetScaler appliance applies the routing table for normal destination-based routing.

srcMac

MAC address to match against the source MAC address of an outgoing IPv4 packet.

protocol

The protocol number in IP header or name.

protocolNumber

The protocol number in IP header or name.

srcPortVal

Port number or range of port numbers to match against the source port number of an outgoing IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

Note: The destination port can be specified only for TCP and UDP protocols.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

destPortVal

Port number or range of port numbers to match against the destination port number of an outgoing IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

Note: The destination port can be specified only for TCP and UDP protocols.

srcIPVal

IP address or range of IP addresses to match against the source IP address of an outgoing IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [10.102.29.30-10.102.29.189].

destIPVal

IP address or range of IP addresses to match against the destination IP address of an outgoing IPv4 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [10.102.29.30-10.102.29.189].

vlan

ID of the VLAN. The NetScaler appliance compares the PBR only to the outgoing packets on the specified VLAN. If you do not specify any interface ID, the appliance compares the PBR to the outgoing packets on all VLANs.

vxlan

ID of the VXLAN. The NetScaler appliance compares the PBR only to the outgoing packets on the specified VXLAN. If you do not specify any interface ID, the appliance compares the PBR to the outgoing packets on all VXLANs.

state

If this route is UP/DOWN.

interface

ID of an interface. The NetScaler appliance compares the PBR only to the outgoing packets on the specified interface. If you do not specify any value, the appliance compares the PBR to the outgoing packets on all interfaces.

hits

The hits of this PBR.

priority

Priority of the PBR, which determines the order in which it is evaluated relative to the other PBRs. If you do not specify priorities while creating PBRs, the PBRs are evaluated in the order in which they are created.

operator

Logical operator.

kernelstate

The commit status of the PBR.

nextHopVal

The Next Hop IP address or gateway name.

ipTunnelName

The iptunnel name where packets need to be forwarded upon.

msr

Whether Monitored Static Route(MSR) is enabled or disabled.

monitor

Name of the monitor, of type PING or ARP, configured on the NetScaler appliance to monitor the route specified by the Next Hop parameter. You must enable the MSR parameter before setting this parameter.

totalprobes

The total number of probes sent.

totalfailedprobes

The total number of failed probes.

failedprobes

Number of the current failed monitoring probes.

monStatCode

The code indicating the monitor response.

monStatParam1

First parameter for use with message code.

monStatParam2

Second parameter for use with message code.

monStatParam3

Third parameter for use with message code.

data

Internal data of this route.

devno

count

stateflag

Example

```
show ns pbr a      Name: a      Action: ALLOW      Hits: 0      src:
```

ns pbr6

The following operations can be performed on "ns pbr6":

[add](#) | [renumber](#) | [rm](#) | [set](#) | [unset](#) | [enable](#) | [disable](#) | [stat](#) | [show](#) | [clear](#) | [apply](#)

add ns pbr6

Adds an IPv6 policy based route (PBR6) to the NetScaler appliance. To commit this operation, you must apply the PBR6s. A PBR6 specifies criteria for selecting outgoing IPv6 packets and, typically, a next hop to which to send the selected packets. For example, you can configure the NetScaler appliance to route outgoing packets from a specific IP address or range to a particular next hop router. Note: The NetScaler appliance process PBR6s before processing the RNAT rules.

Synopsys

```
add ns pbr6 <name> [-td <positive_integer>] <action> [-srcIPv6 [<operator>] <srcIPv6Val>] [-srcPort [<operator>] <srcPortVal>] [-destIPv6 [<operator>] <destIPv6Val>] [-destPort [<operator>] <destPortVal>] [-srcMac <mac_addr>] [-protocol <protocol>] | -protocolNumber <positive_integer>] [-vlan <positive_integer>] | -vxlan <positive_integer>] [-interface <interface_name>] [-priority <positive_integer>] [-state ( ENABLED | DISABLED )] [-msr ( ENABLED | DISABLED )] [-monitor <string>]] [-nextHop <nextHopVal>] [-nextHopVlan <positive_integer>]
```

Arguments

name

Name for the PBR6. Must begin with an ASCII alphabetic or underscore \(_\) character, and must contain only ASCII alphanumeric, underscore, hash \(\#\), period \(\.\), space, colon \(\:\), at \(\@\), equals \(\=\), and hyphen \(\-\) characters. Can be changed after the PBR6 is created.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

action

Action to perform on the outgoing IPv6 packets that match the PBR6.

Available settings function as follows:

- * ALLOW - The NetScaler appliance sends the packet to the designated next-hop router.
- * DENY - The NetScaler appliance applies the routing table for normal destination-based routing.

Possible values: ALLOW, DENY

srcIPv6

IP address or range of IP addresses to match against the source IP address of an outgoing IPv6 packet. In the command line interface, separate the range with a hyphen and enclose within brackets.

operator

Logical operator.

Possible values: =, !=, EQ, NEQ

srcIPv6Val

IP address or range of IP addresses to match against the source IP address of an outgoing IPv6 packet. In the command line interface, separate the range with a hyphen and enclose within brackets.

srcPort

Port number or range of port numbers to match against the source port number of an outgoing IPv6 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

srcPortVal

Source port (range).

Maximum value: 65535

destIPv6

IP address or range of IP addresses to match against the destination IP address of an outgoing IPv6 packet. In the command line interface, separate the range with a hyphen and enclose within brackets.

destIPv6Val

IP address or range of IP addresses to match against the destination IP address of an outgoing IPv6 packet. In the command line interface, separate the range with a hyphen and enclose within brackets.

destPort

Port number or range of port numbers to match against the destination port number of an outgoing IPv6 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

Note: The destination port can be specified only for TCP and UDP protocols.

destPortVal

Destination port (range).

Maximum value: 65535

srcMac

MAC address to match against the source MAC address of an outgoing IPv6 packet.

protocol

Protocol, identified by protocol name, to match against the protocol of an outgoing IPv6 packet.

Possible values: ICMPV6, TCP, UDP

protocolNumber

Protocol, identified by protocol number, to match against the protocol of an outgoing IPv6 packet.

Minimum value: 1

Maximum value: 255

vlan

ID of the VLAN. The NetScaler appliance compares the PBR6 only to the outgoing packets on the specified VLAN. If you do not specify an interface ID, the appliance compares the PBR6 to the outgoing packets on all VLANs.

Minimum value: 1

Maximum value: 4094

vxlan

ID of the VXLAN. The NetScaler appliance compares the PBR6 only to the outgoing packets on the specified VXLAN. If you do not specify an interface ID, the appliance compares the PBR6 to the outgoing packets on all VXLANs.

Minimum value: 1

Maximum value: 16777215

interface

ID of an interface. The NetScaler appliance compares the PBR6 only to the outgoing packets on the specified interface. If you do not specify a value, the appliance compares the PBR6 to the outgoing packets on all interfaces.

priority

Priority of the PBR6, which determines the order in which it is evaluated relative to the other PBR6s. If you do not specify priorities while creating PBR6s, the PBR6s are evaluated in the order in which they are created.

Minimum value: 1

Maximum value: 80000

state

Enable or disable the PBR6. After you apply the PBR6s, the NetScaler appliance compares outgoing packets to the enabled PBR6s.

Possible values: ENABLED, DISABLED

Default value: ENABLED

msr

Monitor the route specified by the Next Hop parameter.

Possible values: ENABLED, DISABLED

Default value: DISABLED

monitor

The name of the monitor.(Can be only of type ping or ARP)

nextHop

IP address of the next hop router to which to send matching packets if action is set to ALLOW. This next hop should be directly reachable from the appliance.

nextHopVal

The Next Hop IPv6 address.

nextHopVlan

VLAN number to be used for link local nexthop .

Minimum value: 1

Maximum value: 4094

Example

```
add ns pbr6 rule1 ALLOW -srcport 45-1024 -destIPv6 2001::45 -nexthop 2001::49
```

renumber ns pbr6

Renumbers the priorities of PBR6s to multiples of 10.To commit this operation, you must apply the PBR6s. Enables you to assign a new PBR6 a priority that is between two existing, consecutively numbered priorities. For example, if two PBR6s, PBR6-1 and PBR6-2, have priorities 2 and 3 renumbering changes those priorities to 20 and 30. You can then add PBR6-3 with priority 25.

Synopsys

```
renumber ns pbr6
```

Example

```
renumber pbr6
```

rm ns pbr6

Removes a PBR6 from the NetScaler appliance. To commit this operation, you must apply the PBR6s.

Synopsys

```
rm ns pbr6 <name> ...
```

Arguments

name

Name of the PBR6 that you want to remove.

Example

```
rm ns pbr6 rule1
```

set ns pbr6

Modifies the specified parameters of a PBR6. To commit this operation, you must apply the PBR6s.

Synopsys

```
set ns pbr6 <name> [-action ( ALLOW | DENY )] [-srcIPv6 [<operator>] <srcIPv6Val>] [-srcPort [<operator>] <srcPortVal>] [-destIPv6 [<operator>] <destIPv6Val>] [-destPort [<operator>] <destPortVal>] [-srcMac <mac_addr>] [-protocol <protocol> | -protocolNumber <positive_integer>] [-vlan <positive_integer> | -vxlan <positive_integer>] [-interface <interface_name>] [-priority <positive_integer>] [-msr ( ENABLED | DISABLED )] [-monitor <string>]] [-nextHop <nextHopVal>] [-nextHopVlan <positive_integer>]
```

Arguments

name

Name of the PBR6 whose parameters you want to modify.

action

Action to perform on the outgoing IPv6 packets that match the PBR6.

Available settings function as follows:

- * ALLOW - The NetScaler appliance sends the packet to the designated next-hop router.
- * DENY - The NetScaler appliance applies the routing table for normal destination-based routing.

Possible values: ALLOW, DENY

srcIPv6

IP address or range of IP addresses to match against the source IP address of an outgoing IPv6 packet. In the command line interface, separate the range with a hyphen and enclose within brackets.

operator

Logical operator.

Possible values: =, !=, EQ, NEQ

srcIPv6Val

IP address or range of IP addresses to match against the source IP address of an outgoing IPv6 packet. In the command line interface, separate the range with a hyphen and enclose within brackets.

srcPort

Source Port (range).

srcPortVal

Port number or range of port numbers to match against the source port number of an outgoing IPv6 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

Maximum value: 65535

destIPv6

IP address or range of IP addresses to match against the destination IP address of an outgoing IPv6 packet. In the command line interface, separate the range with a hyphen and enclose within brackets.

destIPv6Val

IP address or range of IP addresses to match against the destination IP address of an outgoing IPv6 packet. In the command line interface, separate the range with a hyphen and enclose within brackets.

destPort

Destination Port (range).

destPortVal

Port number or range of port numbers to match against the destination port number of an outgoing IPv6 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

Note: The destination port can be specified only for TCP and UDP protocols.

Maximum value: 65535

srcMac

MAC address to match against the source MAC address of an outgoing IPv6 packet.

protocol

Protocol, identified by protocol name, to match against the protocol of an outgoing IPv6 packet.

Possible values: ICMPV6, TCP, UDP

protocolNumber

Protocol, identified by protocol number, to match against the protocol of an outgoing IPv6 packet.

Minimum value: 1

Maximum value: 255

vlan

ID of the VLAN. The NetScaler appliance compares the PBR6 only to the outgoing packets on the specified VLAN. If you do not specify an interface ID, the appliance compares the PBR6 to the outgoing packets on all VLANs.

Minimum value: 1

Maximum value: 4094

vxlan

ID of the VXLAN. The NetScaler appliance compares the PBR6 only to the outgoing packets on the specified VXLAN. If you do not specify an interface ID, the appliance compares the PBR6 to the outgoing packets on all VXLANs.

Minimum value: 1

Maximum value: 16777215

interface

ID of an interface. The NetScaler appliance compares the PBR6 only to the outgoing packets on the specified interface. If you do not specify a value, the appliance compares the PBR6 to the outgoing packets on all interfaces.

priority

Priority of the PBR6, which determines the order in which it is evaluated relative to the other PBR6s. If you do not specify priorities while creating PBR6s, the PBR6s are evaluated in the order in which they are created.

Minimum value: 1

Maximum value: 80000

msr

Monitor the route specified by the Next Hop parameter.

Possible values: ENABLED, DISABLED

Default value: DISABLED

monitor

The name of the monitor.(Can be only of type ping or ARP)

nextHop

IP address of the next hop router to which to send matching packets if action is set to ALLOW. This next hop should be directly reachable from the appliance.

nextHopVal

The Next Hop IPv6 address

nextHopVlan

VLAN number to be used for link local nexthop .

Minimum value: 1

Maximum value: 4094

Example

```
set ns pbr6 rule1 -srcPort 50
```

unset ns pbr6

Resets the attributes of the specified PBR6. Attributes for which a default value is available revert to their default values. Refer to the set ns pbr6 command for descriptions of the parameters..Refer to the set ns pbr6 command for meanings of the arguments.

Synopsis

```
unset ns pbr6 <name> [-srcIPv6] [-srcPort] [-destIPv6] [-destPort] [-srcMac] [-protocol] [-interface] [-vlan] [-vxlan] [-msr] [-monitor] [-nextHop] [-nextHopVlan]
```

Example

```
unset ns pbr6 rule1 -srcPort
```

enable ns pbr6

Enables a PBR6. To commit this operation, you must apply the PBR6s. After you apply the PBR6s, the NetScaler appliance compares outgoing packets to the enabled PBR6.

Synopsys

```
enable ns pbr6 <name> ...
```

Arguments

name

Name of PBR6 that you want to enable.

Example

```
enable ns pbr6 rule1
```

disable ns pbr6

Disables a PBR6. To commit this operation, you must apply the PBR6s. After you apply the PBR6s, the NetScaler appliance does not compare outgoing packets to the disabled PBR6s.

Synopsys

```
disable ns pbr6 <name> ...
```

Arguments

name

Name of PBR6 that you want to disable.

Example

```
disable ns pbr6 rule1
```

stat ns pbr6

Displays statistics related to the PBR6s. To display statistics of all the PBR6s, run the command without any parameters. To display statistics of a particular PBR6, specify the name of the PBR6.

Synopsys

```
stat ns pbr6 [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full )]
```

Arguments

name

Name of the PBR6 whose statistics you want the NetScaler appliance to display.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Allow PBR6 hits (PBR6Allow)

Total packets that matched the PBR6 with action ALLOW

Deny PBR6 hits (PBR6Deny)

Total packets that matched PBR6 with action DENY

PBR6 hits (PBR6TotHits)

Total packets that matched one of the configured PBR6

PBR6 misses (PBR6Miss)

Total packets that did not match any PBR6

Hits for this PBR6 (PBR6Hits)

Number of times the pbr6 was hit

Example

```
stat pbr6
```

show ns pbr6

Displays settings related to the PBR6s. To display settings of all the PBR6s, run the command without any parameters. To display settings of a particular PBR6, specify the name of the PBR6.

Synopsys

```
show ns pbr6 [<name>] [-detail]
```

Arguments

name

Name of the PBR6 whose settings you want the NetScaler appliance to display.

detail

To get a detailed view.

Outputs

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

action

Action to perform on the outgoing IPv6 packets that match the PBR6.

Available settings function as follows:

- * ALLOW - The NetScaler appliance sends the packet to the designated next-hop router.
- * DENY - The NetScaler appliance applies the routing table for normal destination-based routing.

srcMac

MAC address to match against the source MAC address of an outgoing IPv6 packet.

stateflag

PBR6 state flag.

protocol

Protocol number in IPv6 header or name.

protocolNumber

Protocol number in IPv6 header or name.

srcPortVal

Port number or range of port numbers to match against the source port number of an outgoing IPv6 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

destPortVal

Port number or range of port numbers to match against the destination port number of an outgoing IPv6 packet. In the command line interface, separate the range with a hyphen and enclose within brackets. For example: [40-90].

Note: The destination port can be specified only for TCP and UDP protocols.

srcIPv6Val

IP address or range of IP addresses to match against the source IP address of an outgoing IPv6 packet. In the command line interface, separate the range with a hyphen and enclose within brackets.

destIPv6Val

IP address or range of IP addresses to match against the destination IP address of an outgoing IPv6 packet. In the command line interface, separate the range with a hyphen and enclose within brackets.

vlan

ID of the VLAN. The NetScaler appliance compares the PBR6 only to the outgoing packets on the specified VLAN. If you do not specify an interface ID, the appliance compares the PBR6 to the outgoing packets on all VLANs.

vxlan

ID of the VXLAN. The NetScaler appliance compares the PBR6 only to the outgoing packets on the specified VXLAN. If you do not specify an interface ID, the appliance compares the PBR6 to the outgoing packets on all VXLANs.

state

If this route is UP/DOWN.

kernelstate

Commit status of the PBR6.

interface

ID of an interface. The NetScaler appliance compares the PBR6 only to the outgoing packets on the specified interface. If you do not specify a value, the appliance compares the PBR6 to the outgoing packets on all interfaces.

hits

Number of hits of this PBR6.

priority

Priority of the PBR6, which determines the order in which it is evaluated relative to the other PBR6s. If you do not specify priorities while creating PBR6s, the PBR6s are evaluated in the order in which they are created.

operator

Logical operator.

msr

Whether Monitored Static Route(MSR) is enabled or disabled.

monitor

Name of the monitor, of type PING6 or ARP, configured on the NetScaler appliance to monitor the route specified by the Next Hop parameter. You must enable the MSR parameter before setting the Monitor parameter.

totalprobes

The total number of probes sent.

totalfailedprobes

The total number of failed probes.

failedprobes

Number of the current failed monitoring probes.

monStatCode

The code indicating the monitor response.

monStatParam1

First parameter for use with message code.

monStatParam2

Second parameter for use with message code.

monStatParam3

Third parameter for use with message code.

nextHopVal

ID of the VLAN, if you have specified a link local address for the Next Hop parameter.

nextHopVlan

VLAN number to be used for link local nexthop .

data

Internal data of this route.

devno

count

Example

```
show ns pbr6 rule1 1)          Name: r1          Action: DENY          srcIPv4
```

clear ns pbr6

Removes all PBR6s from the NetScaler appliance. This operation does not require an explicit apply.

Synopsys

```
clear ns pbr6
```

Example

```
clear ns pbr6
```

apply ns pbr6

Updates the PBR6's memory tree (lookup table), adding any new PBR6 and applying any modifications to the existing PBR6s. The lookup table includes the configuration of all the extended PBR6s on the NetScaler appliance. The NetScaler appliance uses the lookup table (not the configuration file) to filter the outgoing IPv6 packets.

Synopsys

```
apply ns pbr6
```

Example

```
apply ns pbr6
```

ns pbrs

The following operations can be performed on "ns pbrs":

[renumber](#) | [clear](#) | [apply](#)

renumber ns pbrs

Renumbers the priorities of PBRs to multiples of 10. To commit this operation, you must apply the PBRs. Enables you to assign a new PBR a priority that is between two existing, consecutively numbered priorities. For example, if two PBRs, PBR1 and PBR2, have priorities 2 and 3 renumbering changes those priorities to 20 and 30. You can then add PBR3 with priority 25.

Synopsys

```
renumber ns pbrs
```

Example

```
renumber pbrs
```

clear ns pbrs

Removes all PBRs from the NetScaler appliance. This operation does not require an explicit apply.

Synopsys

```
clear ns pbrs
```

Example

```
clear ns pbrs
```

apply ns pbrs

Updates the PBR's memory tree (lookup table), adding any new PBR and applying any modifications to existing PBRs. The lookup table includes the configuration of all the extended PBRs on the NetScaler appliance. The NetScaler appliance uses the lookup table (not the configuration file) to filter the outgoing IPv4 packets.

Synopsys

```
apply ns pbrs
```

Example

```
apply ns pbrs
```

ns persistence session

The following operations can be performed on "ns persistence session":

[show](#) | [clear](#)

show ns persistence session

Get all Sessions corresponding to a Vserver NOTE: This command is deprecated.Moved to LB command group

Synopsys

Arguments

name

The name of the virtual server.

Outputs

type

The netmask of this IP.

srcIP

SOURCE IP.

srcIPv6

SOURCE IPv6 ADDRESS.

destIP

DESTINATION IP.

destIPv6

DESTINATION IPv6 ADDRESS.

flags

IPv6 FLAGS.

destPort

Destination port.

vServerName

Virtual server name.

timeout

Persistent Session timeout.

referenceCount

Reference Count.

sipCallID

SIP CALLID.

persistenceParam

Specific persistence information . Callid in case of SIP_CALLID persistence entry , RTSP session id in case of RTSP_SESSIONID persistence entry.

devno

count

stateflag

Example

```
show ns persistenceSession vipname
```

clear ns persistenceSession

Use this command to clear/flush persistence sessions NOTE: This command is deprecated.Moved to LB command group

Synopsys

Arguments

vServer

The name of the LB vserver whose persistence sessions are to be flushed. If not specified, all persistence sessions will be flushed .

Example

```
clear persistenceSessions -vserver vip1
```

ns rateControl

The following operations can be performed on "ns rateControl":

[set](#) | [unset](#) | [show](#)

set ns rateControl

Sets the UDP/TCP/ICMP packet rate controls for any application that is not configured at System (direct access to the backend through System).

Synopsys

```
set ns rateControl [-tcpThreshold <positive_integer>] [-udpThreshold <positive_integer>] [-icmpThreshold <positive_integer>] [-tcprstThreshold <positive_integer>]
```

Arguments

tcpThreshold

Number of SYNs permitted per 10 milliseconds.

Minimum value: 0

udpThreshold

Number of UDP packets permitted per 10 milliseconds.

Minimum value: 0

icmpThreshold

Number of ICMP packets permitted per 10 milliseconds.

Default value: 100

Minimum value: 0

tcprstThreshold

The number of TCP RST packets permitted per 10 milli second. zero means rate control is disabled and 0xffffffff means every thing is rate controlled

Default value: 100

Minimum value: 0

Example

The following command will set the SYN rate to 100, icmp rate to 10 and the udp rate to

unset ns rateControl

Use this command to remove ns rateControl settings. Refer to the set ns rateControl command for meanings of the arguments.

Synopsys

```
unset ns rateControl [-tcpThreshold] [-udpThreshold] [-icmpThreshold] [-tcprstThreshold]
```

show ns rateControl

Displays the values configured for rate control on the appliance.

Synopsys

show ns rateControl

Outputs

tcpThreshold

Number of SYNs permitted per 10 milliseconds.

udpThreshold

Number of UDP packets permitted per 10 milliseconds.

icmpThreshold

Number of ICMP packets permitted per 10 milliseconds.

tcprstThreshold

The number of TCP RST packets permitted per 10 milli second. zero means rate control is disabled and 0xffffffff means every thing is rate controlled

Example

By default, there is no rate control for TCP/UDP and for ICMP it will be 100. The output

ns rollbackcmd

The following operations can be performed on "ns rollbackcmd":

show ns rollbackcmd

Generates the command(s) that can be used to roll back the command(s) that are specified in an input file. For example, if you want to roll back the creation of a load balancing virtual server named vserver_test, you must include the 'add lb vserver vserver_test ..' command in the input file. The output of this command is the 'rm lb vserver vserver_test' command.

Synopsys

```
show ns rollbackcmd [-fileName <input_filename>] [-outtype ( cli | xml )]
```

Arguments

fileName

File that contains the commands for which the rollback commands must be generated. Specify the full path of the file name.

outtype

Format in which the rollback commands must be generated.

Possible values: cli, xml

Example

```
show ns rollbackcmd -file <file_name>
```

ns rpcNode

The following operations can be performed on "ns rpcNode":

set | unset | show

set ns rpcNode

Sets the authentication attributes associated with peer system node. All system nodes use Remote Procedure Calls (RPC) to communicate.

Synopsys

```
set ns rpcNode <IPAddress> {-password } [-srcIP <ip_addr|ipv6_addr|*>] [-secure ( YES | NO )]
```

Arguments

IPAddress

IP address of the node. This has to be in the same subnet as the NSIP address.

password

Password to be used in authentication with the peer system node.

srcIP

Source IP address to be used to communicate with the peer system node. The default value is 0, which means that the appliance uses the NSIP address as the source IP address.

secure

State of the channel when talking to the node.

Possible values: YES, NO

Example

Example-1: Failover configuration In a failover configuration define peer NS as: add 1

```
unset ns rpcNode
```

Use this command to remove ns rpcNode settings. Refer to the set ns rpcNode command for meanings of the arguments.

Synopsys

```
unset ns rpcNode <IPAddress> [-password] [-srcIP] [-secure]
```

show ns rpcNode

Display a list of nodes currently communicating by using Remote Procedure Calls (RPC).

Synopsys

```
show ns rpcNode [<IPAddress>]
```

Arguments

IPAddress

IP address of the node.

Outputs

password

Password.

srcIP

The src ip used in communication with the peer System node.

secure

State of the channel when talking to the node.

flags

Flags related to this rpcNode

devno

count

stateflag

Example

Following example shows list of nodes communicating using RPC: `> sh rpcnode 1) IPAddress:`

ns runningConfig

The following operations can be performed on "ns runningConfig":

show ns runningConfig

Displays all the configurations that have been executed on the appliance, including the configurations that have not yet been saved. Note: The unsaved configurations are lost when the appliance is rebooted or shut down.

Synopsys

show ns runningConfig [-withDefaults]

Arguments

withDefaults

Include default values of parameters that have not been explicitly configured. If this argument is disabled, such parameters are not included.

Outputs

response

running config data as text blob

ns savedConfig

The following operations can be performed on "ns savedConfig":

show ns savedConfig

Displays the saved configurations.

Synopsys

show ns savedConfig

Outputs

textBlob

Text of the last saved configuration.

ns simpleacl

The following operations can be performed on "ns simpleacl":

add | **clear** | **rm** | **flush** | **show** | **stat**

add ns simpleacl

Adds a simple ACL rule to the NetScaler appliance. Simple ACL rules filter IPv4 packets on the basis of their source IP addresses and, optionally, the destination port and/or protocol. Any packet with the characteristics specified in the simple ACL rule is dropped.

Synopsys

```
add ns simpleacl <aclname> <aclaction> [-td <positive_integer>] -srcIP <ip_addr> [-destPort <port> -protocol ( TCP  
| UDP )] [-TTL <positive_integer>]
```

Arguments

aclname

Name for the simple ACL rule. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the simple ACL rule is created.

aclaction

Drop incoming IPv4 packets that match the simple ACL rule.

Possible values: DENY

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

srcIP

IP address to match against the source IP address of an incoming IPv4 packet.

destPort

Port number to match against the destination port number of an incoming IPv4 packet.

Omitting the port number creates an all-ports simple ACL rule, which matches any port. In that case, you cannot create another simple ACL rule specifying a specific port and the same source IPv4 address.

protocol

Protocol to match against the protocol of an incoming IPv4 packet. You must set this parameter if you have set the Destination Port parameter.

Possible values: TCP, UDP

TTL

Number of seconds, in multiples of four, after which the simple ACL rule expires. If you do not want the simple ACL rule to expire, do not specify a TTL value.

Minimum value: 4

Maximum value: 2147483647

Example

```
add simpleacl rule1 DENY -srcIP 1.1.1.1 -destPort 80 -protocol TCP add simpleacl rule2 DENY
```

clear ns simpleacl

Removes all simple ACL rules from the NetScaler appliance.

Synopsys

```
clear ns simpleacl
```

rm ns simpleacl

Removes a simple ACL rule from the NetScaler appliance.

Synopsys

```
rm ns simpleacl <aclname> ...
```

Arguments

aclname

Name of the simple ACL rule that you want to remove.

Example

```
rm ns simpleacl rule1
```

flush ns simpleacl

Terminates all established IPv4 connections that match any of the newly configured simple ACL rules. Note: If you plan to create more than one simple ACL rule and flush existing connections that match any of them, you can minimize the affect on performance by first creating all of the simple ACL rules and then running flush only once.

Synopsys

```
flush ns simpleacl -estSessions
```

Arguments

estSessions

show ns simpleacl

Displays settings of all the simple ACL rules or of the specified simple ACL rule. To display settings of all the simple ACL rules, run the command without any parameters. To display settings of a particular simple ACL rule, specify the name of the simple ACL rule.

Synopsys

```
show ns simpleacl [<aclname>]
```

Arguments

aclname

Name of the simple ACL rule whose details you want the NetScaler appliance to display.

Outputs

aclaction

Drop incoming IPv4 packets that match the simple ACL rule.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

srcIP

Source IP address.

stateflag

SACL state flag.

destPort

Destination Port.

protocol

Protocol associated with the ACL rule.

TTL

Time to expire this ACL rule(in seconds).

time

Time when this acl is added.

hits

Number of hits for this ACL rule.

devno**count**

Example

```
show simpleacl rule1      Name: rule1                               Action: DENY  srcIP = 10.102
```

stat ns simpleacl

Displays statistics related to the simple ACL rules.

Synopsys

```
stat ns simpleacl [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

SimpleACL hits (SACLHits)

Packets matching a SimpleACL.

SimpleACL misses (SACLMiss)

Packets not matching any SimpleACL.

SimpleACLs count (SACLsCount)

Number of SimpleACLs configured.

Allow SimpleACL hits (SACLAllow)

Total packets that matched a SimpleACL with action ALLOW and got consumed by NetScaler.

Bridge SimpleACL hits (SACLBdg)

Total packets that matched a SimpleACL with action BRIDGE and got bridged by NetScaler.

Deny SimpleACL hits (SACLDeny)

Packets dropped because they match SimpleACL (Access Control List) with processing mode set to DENY.

Example

```
stat simpleacl
```

ns simpleacl6

The following operations can be performed on "ns simpleacl6":

add | **clear** | **flush** | **rm** | **show** | **stat**

add ns simpleacl6

Adds a simple ACL6 rule to the NetScaler appliance. Simple ACL6 rules filter IPv6 packets on the basis of their source IP addresses and, optionally, the destination port and/or protocol. Any packet with the characteristics specified in the simple ACL6 rule is dropped.

Synopsys

```
add ns simpleacl6 <aclname> [-td <positive_integer>] <aclaction> -srcIPv6 <ipv6_addr|null> [-destPort <port> -  
protocol ( TCP | UDP )] [-TTL <positive_integer>]
```

Arguments

aclname

Name for the simple ACL6 rule. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the simple ACL6 rule is created.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

aclaction

Drop incoming IPv6 packets that match the simple ACL6 rule.

Possible values: DENY

srcIPv6

IP address to match against the source IP address of an incoming IPv6 packet.

destPort

Port number to match against the destination port number of an incoming IPv6 packet.

Omitting the port number creates an all-ports simple ACL6 rule, which matches any port. In that case, you cannot create another simple ACL6 rule specifying a specific port and the same source IPv6 address.

protocol

Protocol to match against the protocol of an incoming IPv6 packet. You must set this parameter if you set the Destination Port parameter.

Possible values: TCP, UDP

TTL

Number of seconds, in multiples of four, after which the simple ACL6 rule expires. If you do not want the simple ACL6 rule to expire, do not specify a TTL value.

Minimum value: 4

Maximum value: 2147483647

Example

```
add simpleacl6 rule1 DENY -srcIP6 fe80::2c0:95ff:fec5:d9b8 -destPort 80 -protocol TCP add
```

clear ns simpleacl6

Removes all simple ACL6 rules from the NetScaler appliance.

Synopsys

```
clear ns simpleacl6
```

Example

```
clear ns simpleacl6
```

flush ns simpleacl6

Terminates all established IPv6 connections that match any of the newly configured simple ACL6 rules. Note: If you plan to create more than one simple ACL6 rule and flush existing connections that match any of them, you can minimize the affect on performance by first creating all of the simple ACL6 rules and then running flush only once.

Synopsys

```
flush ns simpleacl6 -estSessions
```

Arguments

estSessions

rm ns simpleacl6

Removes a simple ACL6 rule from the NetScaler appliance.

Synopsys

```
rm ns simpleacl6 <aclname> ...
```

Arguments

aclname

Name of the simple ACL6 rule that you want to remove.

Example

```
rm ns simpleacl6 rule1
```

show ns simpleacl6

Displays settings of all the simple ACL6 rules or of the specified simple ACL6 rule. To display settings of all the simple ACL6 rules, run the command without any parameters. To display settings of a particular simple ACL6 rule, specify the name of the simple ACL6 rule.

Synopsys

```
show ns simpleacl6 [<aclname>]
```

Arguments

aclname

Name of the simple ACL6 rule whose settings you want the NetScaler appliance to display.

Outputs

aclaction

Action associated with the SACL6 rule.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

srcIPv6

Source IP6 address.

stateflag

SACL6 state flag.

destPort

Destination Port.

protocol

Protocol associated with the ACL rule.

TTL

Time to expire this ACL rule(in seconds).

hits

Number of hits for this SACL6 rule.

time

Time when this acl is added.

devno

count

Example

```
show simpleacl6 rule1 Name: rule1 Action: DENY Hits: 5 src:
```

stat ns simpleacl6

Displays statistics related to the simple ACL6 rules.

Synopsys

```
stat ns simpleacl6 [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

SimpleACL6 hits (SACL6Hits)

Packets matching a SimpleACL6.

SimpleACL6 misses (SACL6Miss)

Packets not matching any SimpleACL6.

SimpleACLs6 count (SACL6sCount)

Number of SimpleACL6s configured.

Allow SimpleACL6 hits (SACL6Allow)

Total packets that matched a SimpleACL6 with action ALLOW and got consumed by NetScaler.

Bridge SimpleACL6 hits (SACL6Bdg)

Total packets that matched a SimpleACL6 with action BRIDGE and got bridged by NetScaler.

Deny SimpleACL6 hits (SACL6Deny)

Packets dropped because they match SimpleACL6 with processing mode set to DENY.

Example

```
stat simpleacl6
```


ns spParams

The following operations can be performed on "ns spParams":

[set](#) | [unset](#) | [show](#)

set ns spParams

Sets surge protection attributes on the appliance.

Synopsys

```
set ns spParams [-baseThreshold <integer>] [-throttle <throttle>]
```

Arguments

baseThreshold

Maximum number of server connections that can be opened before surge protection is activated.

Default value: 200

Maximum value: 32767

throttle

Rate at which the system opens connections to the server.

Possible values: Aggressive, Normal, Relaxed

Default value: Normal

Example

```
set ns spparams -baseThreshold 1000 -throttle aggressive set ns spparams -throttle relaxed
```

unset ns spParams

Use this command to remove ns spParams settings. Refer to the set ns spParams command for meanings of the arguments.

Synopsys

```
unset ns spParams [-baseThreshold] [-throttle]
```

show ns spParams

Displays the surge protection configuration on the appliance. Surge protection parameters are set by using the 'set ns spParams' command.

Synopsys

```
show ns spParams
```

Outputs

baseThreshold

The base threshold. This is the maximum number of server connections that can be open before surge protection is activated.

throttle

Rate at which the system opens connections to the server.

Table

Table.

Example

```
> show ns sparams      Surge Protection parameters:      BaseThreshold: 200      Throttle:  Norma
```

ns stats

The following operations can be performed on "ns stats":

[show](#) | [clear](#)

show ns stats

show ns stats is an alias for stat ns

Synopsys

show ns stats - alias for 'stat ns'

clear ns stats

Clearing stats

Synopsys

clear ns stats <cleanuplevel>

Arguments

cleanuplevel

The level of stats to be cleared. 'global' option will clear global counters only, 'all' option will clear all device counters also along with global counters. For both the cases only 'ever incrementing counters' i.e. total counters will be cleared.

Possible values: global, all

ns surgeQ

The following operations can be performed on "ns surgeQ":

flush ns surgeQ

Flushes the connections that are waiting in SurgeQ. SurgeQ contains the client connections waiting for a server connection.

Synopsys

```
flush ns surgeQ [-name <string> [-serverName <string> <port>]]
```

Arguments

name

Name of a virtual server, service or service group for which the SurgeQ must be flushed.

serverName

Name of a service group member. This argument is needed when you want to flush the SurgeQ of a service group.

port

port on which server is bound to the entity(Servicegroup).

Example

To flush the surgeQ system wide, use the command: `flush ns SurgeQ`. To flush the surgeQ sp

ns tcpParam

The following operations can be performed on "ns tcpParam":

[set](#) | [unset](#) | [show](#)

set ns tcpParam

Sets the TCP parameters for the NetScaler appliance.

Synopsys

```
set ns tcpParam [-WS ( ENABLED | DISABLED )] [-WSVal <positive_integer>] [-SACK ( ENABLED | DISABLED )] [-learnVsvrMSS ( ENABLED | DISABLED )] [-maxBurst <positive_integer>] [-initialCwnd <positive_integer>] [-delayedAck <positive_integer>] [-downStateRST ( ENABLED | DISABLED )] [-nagle ( ENABLED | DISABLED )] [-limitedPersist ( ENABLED | DISABLED )] [-oooQSize <positive_integer>] [-ackOnPush ( ENABLED | DISABLED )] [-maxPktPerMss <integer>] [-pktPerRetx <integer>] [-minRTO <integer>] [-slowStartIncr <integer>] [-maxDynServerProbes <positive_integer>] [-synHoldFastGiveup <positive_integer>] [-maxSynholdPerprobe <positive_integer>] [-maxSynhold <positive_integer>] [-mssLearnInterval <positive_integer>] [-mssLearnDelay <positive_integer>] [-maxTimeWaitConn <positive_integer>] [-maxSynAckRetx <positive_integer>] [-synAttackDetection ( ENABLED | DISABLED )] [-connFlushIfNoMem <connFlushIfNoMem>] [-connFlushThres <positive_integer>] [-mptcpConCloseOnPassiveSF ( ENABLED | DISABLED )] [-mptcpChecksum ( ENABLED | DISABLED )] [-mptcpSFtimeout <secs>] [-mptcpSFReplaceTimeout <secs>] [-mptcpMaxSF <positive_integer>] [-mptcpMaxPendingSF <positive_integer>] [-mptcpPendingJoinThreshold <positive_integer>] [-mptcpRTOsToSwitchSF <positive_integer>] [-mptcpUseBackupOnDSS ( ENABLED | DISABLED )] [-TcpMaxRetries <positive_integer>] [-mptcpImmediateSFCloseOnFIN ( ENABLED | DISABLED )] [-mptcpCloseMptcpSessionOnLastSFClose ( ENABLED | DISABLED )]
```

Arguments

WS

Enable or disable window scaling.

Possible values: ENABLED, DISABLED

Default value: DISABLED

WSVal

Factor used to calculate the new window size.

This argument is needed only when the window scaling is enabled.

Default value: 4

Minimum value: 0

Maximum value: 14

SACK

Enable or disable Selective ACKnowledgement (SACK).

Possible values: ENABLED, DISABLED

Default value: DISABLED

learnVsvrMSS

Enable or disable maximum segment size (MSS) learning for virtual servers.

Possible values: ENABLED, DISABLED

Default value: DISABLED

maxBurst

Maximum number of TCP segments allowed in a burst.

Default value: 6

Minimum value: 1

Maximum value: 255

initialCwnd

Initial maximum upper limit on the number of TCP packets that can be outstanding on the TCP link to the server.

Default value: 4

Minimum value: 1

Maximum value: 44

delayedAck

Timeout for TCP delayed ACK, in milliseconds.

Default value: 100

Minimum value: 10

Maximum value: 300

downStateRST

Flag to switch on RST on down services.

Possible values: ENABLED, DISABLED

Default value: DISABLED

nagle

Enable or disable the Nagle algorithm on TCP connections.

Possible values: ENABLED, DISABLED

Default value: DISABLED

limitedPersist

Limit the number of persist (zero window) probes.

Possible values: ENABLED, DISABLED

Default value: ENABLED

oooQSize

Maximum size of out-of-order packets queue. A value of 0 means no limit.

Default value: 64

Minimum value: 0

Maximum value: 65535

ackOnPush

Send immediate positive acknowledgement (ACK) on receipt of TCP packets when doing Web 2.0 PUSH.

Possible values: ENABLED, DISABLED

Default value: ENABLED

maxPktPerMss

Maximum number of TCP packets allowed per maximum segment size (MSS).

Minimum value: 0

Maximum value: 1460

pktPerRetx

Maximum limit on the number of packets that should be retransmitted on receiving a partial ACK.

Default value: 1

Minimum value: 1

Maximum value: 100

minRTO

Minimum retransmission timeout, in milliseconds, specified in 10-millisecond increments (value must yield a whole number if divided by 10).

Default value: 1000

Minimum value: 10

Maximum value: 64000

slowStartIncr

Multiplier that determines the rate at which slow start increases the size of the TCP transmission window after each acknowledgement of successful transmission.

Default value: 2

Minimum value: 1

Maximum value: 100

maxDynServerProbes

Maximum number of probes that NetScaler can send out in 10 milliseconds, to dynamically learn a service. NetScaler probes for the existence of the origin in case of wildcard virtual server or services.

Default value: 7

Minimum value: 1

Maximum value: 65535

synHoldFastGiveup

Maximum threshold. After crossing this threshold number of outstanding probes for origin, the NetScaler reduces the number of connection retries for probe connections.

Default value: 1024

Minimum value: 256

Maximum value: 65535

maxSynholdPerprobe

Limit the number of client connections (SYN) waiting for status of single probe. Any new SYN packets will be dropped.

Default value: 128

Minimum value: 1

Maximum value: 255

maxSynhold

Limit the number of client connections (SYN) waiting for status of probe system wide. Any new SYN packets will be dropped.

Default value: 16384

Minimum value: 256

Maximum value: 65535

mssLearnInterval

Duration, in seconds, to sample the Maximum Segment Size (MSS) of the services. The NetScaler appliance determines the best MSS to set for the virtual server based on this sampling. The argument to enable maximum segment size (MSS) for virtual servers must be enabled.

Default value: 180

Minimum value: 1

Maximum value: 1048576

mssLearnDelay

Frequency, in seconds, at which the virtual servers learn the Maximum segment size (MSS) from the services. The argument to enable maximum segment size (MSS) for virtual servers must be enabled.

Default value: 3600

Minimum value: 1

Maximum value: 1048576

maxTimeWaitConn

Maximum number of connections to hold in the TCP TIME_WAIT state on a packet engine. New connections entering TIME_WAIT state are proactively cleaned up.

Default value: 7000

Minimum value: 1

maxSynAckRetx

When 'syncookie' is disabled in the TCP profile that is bound to the virtual server or service, and the number of TCP SYN+ACK retransmission by NetScaler for that virtual server or service crosses this threshold, the NetScaler appliance responds by using the TCP SYN-Cookie mechanism.

Default value: 100

Minimum value: 100

Maximum value: 1048576

synAttackDetection

Detect TCP SYN packet flood and send an SNMP trap.

Possible values: ENABLED, DISABLED

Default value: ENABLED

connFlushIfNoMem

Flush an existing connection if no memory can be obtained for new connection.

HALF_CLOSED_AND_IDLE: Flush a connection that is closed by us but not by peer, or failing that, a connection that is past configured idle time. New connection fails if no such connection can be found.

FIFO: If no half-closed or idle connection can be found, flush the oldest non-management connection, even if it is active. New connection fails if the oldest few connections are management connections.

Note: If you enable this setting, you should also consider lowering the zombie timeout and half-close timeout, while setting the NetScaler timeout.

See Also: `connFlushThres` argument below.

Possible values: NONE, HALFCLOSED_AND_IDLE, FIFO

Default value: 5

connFlushThres

Flush an existing connection (as configured through `-connFlushIfNoMem` FIFO) if the system has more than specified number of connections, and a new connection is to be established. Note: This value may be rounded down to be a whole multiple of the number of packet engines running.

Minimum value: 1

mptcpConCloseOnPassiveSF

Accept DATA_FIN/FAST_CLOSE on passive subflow

Possible values: ENABLED, DISABLED

Default value: ENABLED

mptcpChecksum

Use MPTCP DSS checksum

Possible values: ENABLED, DISABLED

Default value: ENABLED

mptcpSFtimeout

The timeout value in seconds for idle mptcp subflows. If this timeout is not set, idle subflows are cleared after `cltTimeout` of `vserver`

Default value: 0

Maximum value: 31536000

mptcpSFReplaceTimeout

The minimum idle time value in seconds for idle mptcp subflows after which the subflow is replaced by new incoming subflow if maximum subflow limit is reached. The priority for replacement is given to those subflow without any transaction

Default value: 10

Maximum value: 31536000

mptcpMaxSF

Maximum number of subflow connections supported in established state per mptcp connection.

Default value: 4

Minimum value: 2

Maximum value: 6

mptcpMaxPendingSF

Maximum number of subflow connections supported in pending join state per mptcp connection.

Default value: 4

Minimum value: 0

Maximum value: 4

mptcpPendingJoinThreshold

Maximum system level pending join connections allowed.

Default value: 0

Minimum value: 0

Maximum value: 4294967294

mptcpRTOsToSwitchSF

Number of RTO's at subflow level, after which MPCTP should start using other subflow.

Default value: 2

Minimum value: 1

Maximum value: 6

mptcpUseBackupOnDSS

When enabled, if NS receives a DSS on a backup subflow, NS will start using that subflow to send data. And if disabled, NS will continue to transmit on current chosen subflow. In case there is some error on a subflow (like RTO's/RST etc.) then NS can choose a backup subflow irrespective of this tunable.

Possible values: ENABLED, DISABLED

Default value: ENABLED

TcpMaxRetries

Number of RTO's after which a connection should be freed.

Default value: 7

Minimum value: 1

Maximum value: 7

mptcpImmediateSFCloseOnFIN

Allow subflows to close immediately on FIN before the DATA_FIN exchange is completed at mptcp level.

Possible values: ENABLED, DISABLED

Default value: DISABLED

mptcpCloseMptcpSessionOnLastSFClose

Allow to send DATA FIN or FAST CLOSE on mptcp connection while sending FIN or RST on the last subflow.

Possible values: ENABLED, DISABLED

Default value: DISABLED

unset ns tcpParam

Use this command to remove ns tcpParam settings. Refer to the set ns tcpParam command for meanings of the arguments.

Synopsis

```
unset ns tcpParam [-WS] [-WSVal] [-SACK] [-learnVsvrMSS] [-maxBurst] [-initialCwnd] [-delayedAck] [-downStateRST] [-nagle] [-limitedPersist] [-oooQSize] [-ackOnPush] [-maxPktPerMss] [-pktPerRetx] [-minRTO] [-slowStartIncr] [-maxDynServerProbes] [-synHoldFastGiveup] [-maxSynholdPerprobe] [-maxSynhold] [-
```

mssLearnInterval] [-mssLearnDelay] [-maxTimeWaitConn] [-maxSynAckRetx] [-synAttackDetection] [-connFlushIfNoMem] [-connFlushThres] [-mptcpConCloseOnPassiveSF] [-mptcpChecksum] [-mptcpSFtimeout] [-mptcpSFReplaceTimeout] [-mptcpMaxSF] [-mptcpMaxPendingSF] [-mptcpPendingJoinThreshold] [-mptcpRTOsToSwitchSF] [-mptcpUseBackupOnDSS] [-TcpMaxRetries] [-mptcpImmediateSFCloseOnFIN] [-mptcpCloseMptcpSessionOnLastSFClose]

show ns tcpParam

Displays the TCP parameters configured on the NetScaler appliance.

Synopsys

show ns tcpParam

Outputs

WS

Enable or disable window scaling.

WSVal

Factor used to calculate the new window size.

This argument is needed only when the window scaling is enabled.

SACK

Enable or disable Selective ACKnowledgement (SACK).

learnVsvrMSS

Enable or disable maximum segment size (MSS) learning for virtual servers.

maxBurst

Maximum number of TCP segments allowed in a burst.

initialCwnd

Initial maximum upper limit on the number of TCP packets that can be outstanding on the TCP link to the server.

recvBuffSize

TCP Receive buffer size

delayedAck

Timeout for TCP delayed ACK, in milliseconds.

downStateRST

Flag to switch on RST on down services.

nagle

Enable or disable the Nagle algorithm on TCP connections.

limitedPersist

Limit the number of persist (zero window) probes.

oooQSize

Maximum size of out-of-order packets queue. A value of 0 means no limit.

ackOnPush

Immediate ACK on PUSH packet

maxPktPerMss

Maximum packets per MSS

pktPerRetx

Maximum packets per retransmission

minRTO

Minimum retransmission timeout, in milliseconds, specified in 10-millisecond increments (value must yield a whole number if divided by 10).

slowStartIncr

TCP slowstart increment factor

maxDynServerProbes

Maximum number of probes that NetScaler can send out in 10 milliseconds, to dynamically learn a service. NetScaler probes for the existence of the origin in case of wildcard virtual server or services.

synHoldFastGiveup

Maximum threshold. After crossing this threshold number of outstanding probes for origin, the NetScaler reduces the number of connection retries for probe connections.

maxSynholdPerprobe

Limit the number of client connections (SYN) waiting for status of single probe. Any new SYN packets will be dropped.

maxSynhold

Limit the number of client connections (SYN) waiting for status of probe system wide. Any new SYN packets will be dropped.

mssLearnInterval

Duration, in seconds, to sample the Maximum Segment Size (MSS) of the services. The NetScaler appliance determines the best MSS to set for the virtual server based on this sampling. The argument to enable maximum segment size (MSS) for virtual servers must be enabled.

mssLearnDelay

Frequency, in seconds, at which the virtual servers learn the Maximum segment size (MSS) from the services. The argument to enable maximum segment size (MSS) for virtual servers must be enabled.

maxTimeWaitConn

Maximum number of connections to hold in the TCP TIME_WAIT state on a packet engine. New connections entering TIME_WAIT state are proactively cleaned up.

KAprrobeUpdateLastactivity

Update last activity for KA probes

maxSynAckRetx

When 'syncookie' is disabled in the TCP profile that is bound to the virtual server or service, and the number of TCP SYN+ACK retransmission by NetScaler for that virtual server or service crosses this threshold, the NetScaler appliance responds by using the TCP SYN-Cookie mechanism.

synAttackDetection

Detect TCP SYN packet flood and send an SNMP trap.

connFlushIfNoMem

Flush an existing connection if no memory can be obtained for new connection.

HALF_CLOSED_AND_IDLE: Flush a connection that is closed by us but not by peer, or failing that, a connection that is past configured idle time. New connection fails if no such connection can be found.

FIFO: If no half-closed or idle connection can be found, flush the oldest non-management connection, even if it is active. New connection fails if the oldest few connections are management connections.

Note: If you enable this setting, you should also consider lowering the zombie timeout and half-close timeout, while setting the NetScaler timeout.

See Also: `connFlushThres` argument below.

connFlushThres

Flush an existing connection (as configured through `-connFlushIfNoMem FIFO`) if the system has more than specified number of connections, and a new connection is to be established. Note: This value may be rounded down to be a whole multiple of the number of packet engines running.

mptcpConCloseOnPassiveSF

Accept `DATA_FIN/FAST_CLOSE` on passive subflow

mptcpChecksum

Use MPTCP DSS checksum

mptcpSFtimeout

The timeout value in seconds for idle mptcp subflows. If this timeout is not set, idle subflows are cleared after `cltTimeout` of vserver

mptcpSFReplaceTimeout

The minimum idle time value in seconds for idle mptcp subflows after which the subflow is replaced by new incoming subflow if maximum subflow limit is reached. The priority for replacement is given to those subflow without any transaction

mptcpMaxSF

Maximum number of subflow connections supported in established state per mptcp connection.

mptcpMaxPendingSF

Maximum number of subflow connections supported in pending join state per mptcp connection.

mptcpPendingJoinThreshold

Maximum system level pending join connections allowed.

mptcpRTOsToSwitchSF

Number of RTO's at subflow level, after which MPCTP should start using other subflow.

mptcpUseBackupOnDSS

When enabled, if NS receives a DSS on a backup subflow, NS will start using that subflow to send data. And if disabled, NS will continue to transmit on current chosen subflow. In case there is some error on a subflow (like RTO's/RST etc.) then NS can choose a backup subflow irrespective of this tunable.

TcpMaxRetries

Number of RTO's after which a connection should be freed.

mptcpImmediateSFCloseOnFIN

Allow subflows to close immediately on FIN before the `DATA_FIN` exchange is completed at mptcp level.

mptcpCloseMptcpSessionOnLastSFClose

Allow to send `DATA FIN` or `FAST CLOSE` on mptcp connection while sending `FIN` or `RST` on the last subflow.

ns tcpProfile

The following operations can be performed on "ns tcpProfile":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add ns tcpProfile

Adds a TCP profile to the NetScaler appliance.

Synopsys

```
add ns tcpProfile <name> [-WS ( ENABLED | DISABLED )] [-SACK ( ENABLED | DISABLED )] [-WSVal  
<positive_integer>] [-nagle ( ENABLED | DISABLED )] [-ackOnPush ( ENABLED | DISABLED )] [-mss  
<positive_integer>] [-maxBurst <positive_integer>] [-initialCwnd <positive_integer>] [-delayedAck <positive_integer>]  
[-oooQSize <positive_integer>] [-maxPktPerMss <positive_integer>] [-pktPerRetx <positive_integer>] [-minRTO  
<positive_integer>] [-slowStartIncr <positive_integer>] [-bufferSize <positive_integer>] [-synCookie ( ENABLED |  
DISABLED )] [-KAprobeUpdateLastactivity ( ENABLED | DISABLED )] [-flavor <flavor>] [-dynamicReceiveBuffering (   
ENABLED | DISABLED )] [-KA ( ENABLED | DISABLED )] [-KAconnIdleTime <positive_integer>] [-KAmassProbes  
<positive_integer>] [-KAprobeInterval <positive_integer>] [-sendBuffsize <positive_integer>] [-mptcp ( ENABLED |  
DISABLED )] [-EstablishClientConn <EstablishClientConn>] [-tcpSegOffload ( AUTOMATIC | DISABLED )] [-  
rstWindowAttenuate ( ENABLED | DISABLED )] [-rstMaxAck ( ENABLED | DISABLED )] [-spoofSynDrop (   
ENABLED | DISABLED )] [-ecn ( ENABLED | DISABLED )] [-mptcpDropDataOnPreEstSF ( ENABLED | DISABLED  
)] [-mptcpFastOpen ( ENABLED | DISABLED )] [-mptcpSessionTimeout <positive_integer>] [-TimeStamp (   
ENABLED | DISABLED )] [-dsack ( ENABLED | DISABLED )] [-ackAggregation ( ENABLED | DISABLED )] [-frto (   
ENABLED | DISABLED )]
```

Arguments

name

Name for a TCP profile. Must begin with a letter, number, or the underscore \(_\) character. Other characters allowed, after the first character, are the hyphen \(-\), period \(. \), hash \(\#\), space \(\), at \(@\), and equal \(\=\) characters. The name of a TCP profile cannot be changed after it is created.

CLI Users: If the name includes one or more spaces, enclose the name in double or single quotation marks \("my tcp profile" or 'my tcp profile'\).

WS

Enable or disable window scaling.

Possible values: ENABLED, DISABLED

Default value: DISABLED

SACK

Enable or disable Selective ACKnowledgement (SACK).

Possible values: ENABLED, DISABLED

Default value: DISABLED

WSVal

Factor used to calculate the new window size.

This argument is needed only when window scaling is enabled.

Default value: 4

Minimum value: 0

Maximum value: 14

nagle

Enable or disable the Nagle algorithm on TCP connections.

Possible values: ENABLED, DISABLED

Default value: DISABLED

ackOnPush

Send immediate positive acknowledgement (ACK) on receipt of TCP packets when doing Web 2.0 PUSH.

Possible values: ENABLED, DISABLED

Default value: ENABLED

mss

Maximum number of octets to allow in a TCP data segment.

Minimum value: 0

Maximum value: 9176

maxBurst

Maximum number of TCP segments allowed in a burst.

Default value: 6

Minimum value: 1

Maximum value: 255

initialCwnd

Initial maximum upper limit on the number of TCP packets that can be outstanding on the TCP link to the server.

Default value: 4

Minimum value: 1

Maximum value: 44

delayedAck

Timeout for TCP delayed ACK, in milliseconds.

Default value: 100

Minimum value: 10

Maximum value: 300

oooQSize

Maximum size of out-of-order packets queue. A value of 0 means no limit.

Default value: 64

Minimum value: 0

Maximum value: 65535

maxPktPerMss

Maximum number of TCP packets allowed per maximum segment size (MSS).

Minimum value: 0

Maximum value: 1460

pktPerRetx

Maximum limit on the number of packets that should be retransmitted on receiving a partial ACK.

Default value: 1

Minimum value: 1

Maximum value: 512

minRTO

Minimum retransmission timeout, in milliseconds.

Default value: 1000

Minimum value: 10

Maximum value: 64000

slowStartIncr

Multiplier that determines the rate at which slow start increases the size of the TCP transmission window after each acknowledgement of successful transmission.

Default value: 2

Minimum value: 1

Maximum value: 100

bufferSize

TCP buffering size, in bytes.

Default value: 8190

Minimum value: 8190

Maximum value: 4194304

synCookie

Enable or disable the SYNCOOKIE mechanism for TCP handshake with clients. Disabling SYNCOOKIE prevents SYN attack protection on the NetScaler appliance.

Possible values: ENABLED, DISABLED

Default value: ENABLED

KAprrobeUpdateLastactivity

Update last activity for the connection after receiving keep-alive (KA) probes.

Possible values: ENABLED, DISABLED

Default value: ENABLED

flavor

Set TCP congestion control algorithm.

Possible values: Default, Westwood, BIC, CUBIC

Default value: Default

dynamicReceiveBuffering

Enable or disable dynamic receive buffering. When enabled, allows the receive buffer to be adjusted dynamically based on memory and network conditions.

Note: The buffer size argument must be set for dynamic adjustments to take place.

Possible values: ENABLED, DISABLED

Default value: ENABLED

KA

Send periodic TCP keep-alive (KA) probes to check if peer is still up.

Possible values: ENABLED, DISABLED

Default value: DISABLED

KAconnIdleTime

Duration, in seconds, for the connection to be idle, before sending a keep-alive (KA) probe.

Default value: NSTCP_KA_DEFAULT_CONN_IDLETIME

Minimum value: 1

Maximum value: 4095

KAmaxProbes

Number of keep-alive (KA) probes to be sent when not acknowledged, before assuming the peer to be down.

Default value: NSTCP_KA_DEFAULT_PROBE_COUNT

Minimum value: 1

Maximum value: 255

KAprobelInterval

Time interval, in seconds, before the next keep-alive (KA) probe, if the peer does not respond.

Default value: NSTCP_KA_DEFAULT_INTERVAL

Minimum value: 1

Maximum value: 4095

sendBuffsize

TCP Send Buffer Size

Default value: 8190

Minimum value: 8190

Maximum value: 4194304

mptcp

Enable or disable Multipath TCP.

Possible values: ENABLED, DISABLED

Default value: DISABLED

EstablishClientConn

Establishing Client Client connection on First data/ Final-ACK / Automatic

Possible values: AUTOMATIC, CONN_ESTABLISHED, ON_FIRST_DATA

Default value: AUTOMATIC

tcpSegOffload

Offload TCP segmentation to the NIC. If set to AUTOMATIC, TCP segmentation will be offloaded to the NIC, if the NIC supports it.

Possible values: AUTOMATIC, DISABLED

Default value: AUTOMATIC

rstWindowAttenuate

Enable or disable RST window attenuation to protect against spoofing. When enabled, will reply with corrective ACK when a sequence number is invalid.

Possible values: ENABLED, DISABLED

Default value: DISABLED

rstMaxAck

Enable or disable acceptance of RST that is out of window yet echoes highest ACK sequence number. Useful only in proxy mode.

Possible values: ENABLED, DISABLED

Default value: DISABLED

spoofSynDrop

Enable or disable drop of invalid SYN packets to protect against spoofing. When disabled, established connections will be reset when a SYN packet is received.

Possible values: ENABLED, DISABLED

Default value: ENABLED

ecn

Enable or disable TCP Explicit Congestion Notification.

Possible values: ENABLED, DISABLED

Default value: DISABLED

mptcpDropDataOnPreEstSF

Enable or disable silently dropping the data on Pre-Established subflow. When enabled, DSS data packets are dropped silently instead of dropping the connection when data is received on pre established subflow.

Possible values: ENABLED, DISABLED

Default value: DISABLED

mptcpFastOpen

Enable or disable Multipath TCP fastopen. When enabled, DSS data packets are accepted before receiving the third ack of SYN handshake.

Possible values: ENABLED, DISABLED

Default value: DISABLED

mptcpSessionTimeout

MPTCP session timeout in seconds. If this value is not set, idle MPTCP sessions are flushed after vserver's client idle timeout.

Default value: 0

Minimum value: 0

Maximum value: 86400

TimeStamp

Enable or Disable TCP Timestamp option (RFC 1323)

Possible values: ENABLED, DISABLED

Default value: DISABLED

dsack

Enable or disable DSACK.

Possible values: ENABLED, DISABLED

Default value: ENABLED

ackAggregation

Enable or disable ACK Aggregation.

Possible values: ENABLED, DISABLED

Default value: DISABLED

frto

Enable or disable FRTO (Forward RTO-Recovery).

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
add tcpprofile <profile name> -WS ENABLED -WSVAL 4
```

rm ns tcpProfile

Removes a TCP profile from the appliance.

Synopsis

```
rm ns tcpProfile <name>
```

Arguments

name

Name of the TCP profile to be removed.

Example

```
rm tcpprofile <profile name>
```

set ns tcpProfile

Modifies the attributes of a TCP profile.

Synopsis

```
set ns tcpProfile <name> [-WS ( ENABLED | DISABLED )] [-SACK ( ENABLED | DISABLED )] [-WSVal  
<positive_integer>] [-nagle ( ENABLED | DISABLED )] [-ackOnPush ( ENABLED | DISABLED )] [-mss  
<positive_integer>] [-maxBurst <positive_integer>] [-initialCwnd <positive_integer>] [-delayedAck <positive_integer>]  
[-oooQSize <positive_integer>] [-maxPktPerMss <positive_integer>] [-pktPerRetx <positive_integer>] [-minRTO  
<positive_integer>] [-slowStartIncr <positive_integer>] [-bufferSize <positive_integer>] [-synCookie ( ENABLED |  
DISABLED )] [-KAProbeUpdateLastactivity ( ENABLED | DISABLED )] [-flavor <flavor>] [-dynamicReceiveBuffering (
```

ENABLED | DISABLED)) [-KA (ENABLED | DISABLED)) [-KAconnIdleTime <positive_integer>] [-KAmaxProbes <positive_integer>] [-KAprobeInterval <positive_integer>] [-sendBuffsize <positive_integer>] [-mptcp (ENABLED | DISABLED)) [-EstablishClientConn <EstablishClientConn>] [-tcpSegOffload (AUTOMATIC | DISABLED)) [-rstWindowAttenuate (ENABLED | DISABLED)) [-rstMaxAck (ENABLED | DISABLED)) [-spooofSynDrop (ENABLED | DISABLED)) [-ecn (ENABLED | DISABLED)) [-mptcpDropDataOnPreEstSF (ENABLED | DISABLED)) [-mptcpFastOpen (ENABLED | DISABLED)) [-mptcpSessionTimeout <positive_integer>] [-TimeStamp (ENABLED | DISABLED)) [-dsack (ENABLED | DISABLED)) [-ackAggregation (ENABLED | DISABLED)) [-frto (ENABLED | DISABLED))

Arguments

name

Name of the TCP profile to be modified.

WS

Enable or disable window scaling.

Possible values: ENABLED, DISABLED

Default value: DISABLED

SACK

Enable or disable Selective ACKnowledgement (SACK).

Possible values: ENABLED, DISABLED

Default value: DISABLED

WSVal

Factor used to calculate the new window size.

This argument is needed only when window scaling is enabled.

Default value: 4

Minimum value: 0

Maximum value: 14

nagle

Enable or disable the Nagle algorithm on TCP connections.

Possible values: ENABLED, DISABLED

Default value: DISABLED

ackOnPush

Send immediate positive acknowledgement (ACK) on receipt of TCP packets when doing Web 2.0 PUSH.

Possible values: ENABLED, DISABLED

Default value: ENABLED

mss

Set Maximum Segment Size(MSS) to use for TCP Connection(0 forces use of global setting)

Minimum value: 0

Maximum value: 9176

maxBurst

Maximum number of TCP segments allowed in a burst.

Default value: 6

Minimum value: 1

Maximum value: 255

initialCwnd

Initial maximum upper limit on the number of TCP packets that can be outstanding on the TCP link to the server.

Default value: 4

Minimum value: 1

Maximum value: 44

delayedAck

Timeout for TCP delayed ACK, in milliseconds.

Default value: 100

Minimum value: 10

Maximum value: 300

oooQSize

Maximum size of out-of-order packets queue. A value of 0 means no limit.

Default value: 64

Minimum value: 0

Maximum value: 65535

maxPktPerMss

Maximum number of TCP packets allowed per maximum segment size (MSS).

Minimum value: 0

Maximum value: 1460

pktPerRetx

Maximum limit on the number of packets that should be retransmitted on receiving a partial ACK.

Default value: 1

Minimum value: 1

Maximum value: 512

minRTO

Minimum retransmission timeout, in milliseconds.

Default value: 1000

Minimum value: 10

Maximum value: 64000

slowStartIncr

Multiplier that determines the rate at which slow start increases the size of the TCP transmission window after each acknowledgement of successful transmission.

Default value: 2

Minimum value: 1

Maximum value: 100

bufferSize

TCP buffering size, in bytes.

Default value: 8190

Minimum value: 8190

Maximum value: 4194304

synCookie

Enable or disable the SYNCOOKIE mechanism for TCP handshake with clients. Disabling SYNCOOKIE prevents SYN attack protection on the NetScaler appliance.

Possible values: ENABLED, DISABLED

Default value: ENABLED

KAprobeUpdateLastactivity

Update last activity for the connection after receiving keep-alive (KA) probes.

Possible values: ENABLED, DISABLED

Default value: ENABLED

flavor

Set TCP congestion control algorithm.

Possible values: Default, Westwood, BIC, CUBIC

Default value: Default

dynamicReceiveBuffering

Enable or disable dynamic receive buffering. When enabled, allows the receive buffer to be adjusted dynamically based on memory and network conditions.

Note: The buffer size argument must be set for dynamic adjustments to take place.

Possible values: ENABLED, DISABLED

Default value: ENABLED

KA

Send periodic TCP keep-alive (KA) probes to check if peer is still up.

Possible values: ENABLED, DISABLED

Default value: DISABLED

KAcnnIdleTime

Duration, in seconds, for the connection to be idle, before sending a keep-alive (KA) probe.

Default value: NSTCP_KA_DEFAULT_CONN_IDLETIME

Minimum value: 1

Maximum value: 4095

KAmxProbes

Number of keep-alive (KA) probes to be sent when not acknowledged, before assuming the peer to be down.

Default value: NSTCP_KA_DEFAULT_PROBE_COUNT

Minimum value: 1

Maximum value: 255

KApobelInterval

Time interval, in seconds, before the next keep-alive (KA) probe, if the peer does not respond.

Default value: NSTCP_KA_DEFAULT_INTERVAL

Minimum value: 1

Maximum value: 4095

sendBuffsize

TCP Send Buffer Size

Default value: 8190

Minimum value: 8190

Maximum value: 4194304

mptcp

Enable or disable Multipath TCP.

Possible values: ENABLED, DISABLED

Default value: DISABLED

EstablishClientConn

Establishing Client Client connection on First data/ Final-ACK / Automatic

Possible values: AUTOMATIC, CONN_ESTABLISHED, ON_FIRST_DATA

Default value: AUTOMATIC

tcpSegOffload

Offload TCP segmentation to the NIC. If set to AUTOMATIC, TCP segmentation will be offloaded to the NIC, if the NIC supports it.

Possible values: AUTOMATIC, DISABLED

Default value: AUTOMATIC

rstWindowAttenuate

Enable or disable RST window attenuation to protect against spoofing. When enabled, will reply with corrective ACK when a sequence number is invalid.

Possible values: ENABLED, DISABLED

Default value: DISABLED

rstMaxAck

Enable or disable acceptance of RST that is out of window yet echoes highest ACK sequence number. Useful only in proxy mode.

Possible values: ENABLED, DISABLED

Default value: DISABLED

spoofSynDrop

Enable or disable drop of invalid SYN packets to protect against spoofing. When disabled, established connections will be reset when a SYN packet is received.

Possible values: ENABLED, DISABLED

Default value: ENABLED

ecn

Enable or disable TCP Explicit Congestion Notification.

Possible values: ENABLED, DISABLED

Default value: DISABLED

mptcpDropDataOnPreEstSF

Enable or disable silently dropping the data on Pre-Established subflow. When enabled, DSS data packets are dropped silently instead of dropping the connection when data is received on pre established subflow.

Possible values: ENABLED, DISABLED

Default value: DISABLED

mptcpFastOpen

Enable or disable Multipath TCP fastopen. When enabled, DSS data packets are accepted before receiving the third ack of SYN handshake.

Possible values: ENABLED, DISABLED

Default value: DISABLED

mptcpSessionTimeout

MPTCP session timeout in seconds. If this value is not set, idle MPTCP sessions are flushed after vserver's client idle timeout.

Default value: 0

Minimum value: 0

Maximum value: 86400

TimeStamp

Enable or Disable TCP Timestamp option (RFC 1323)

Possible values: ENABLED, DISABLED

Default value: DISABLED

dsack

Enable or disable DSACK.

Possible values: ENABLED, DISABLED

Default value: ENABLED

ackAggregation

Enable or disable ACK Aggregation.

Possible values: ENABLED, DISABLED

Default value: DISABLED

frto

Enable or disable FRTO (Forward RTO-Recovery).

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
set tcpprofile <profile name> -WS ENABLED -WSVAL 4
```

unset ns tcpProfile

Removes the attributes of the TCP profile. Attributes for which a default value is available revert to their default values. Refer to the 'set ns tcpProfile' command for a description of the parameters..Refer to the set ns tcpProfile command for meanings of the arguments.

Synopsys

```
unset ns tcpProfile <name> [-WS] [-SACK] [-WSVal] [-nagle] [-ackOnPush] [-mss] [-maxBurst] [-initialCwnd] [-delayedAck] [-oooQSize] [-maxPktPerMss] [-pktPerRetx] [-minRTO] [-slowStartIncr] [-bufferSize] [-synCookie] [-KAprobeUpdateLastactivity] [-flavor] [-dynamicReceiveBuffering] [-KA] [-KAmassProbes] [-KAconnIdleTime] [-KAprobeInterval] [-sendBuffsize] [-mptcp] [-EstablishClientConn] [-tcpSegOffload] [-rstWindowAttenuate] [-rstMaxAck] [-spoofSynDrop] [-ecn] [-mptcpDropDataOnPreEstSF] [-mptcpFastOpen] [-mptcpSessionTimeout] [-TimeStamp] [-dsack] [-ackAggregation] [-frto]
```

show ns tcpProfile

Displays information about TCP profiles configured on the appliance.

Synopsys

```
show ns tcpProfile [<name>]
```

Arguments

name

Name of the TCP profile to be displayed. If a name is not provided, information about all TCP profiles is shown.

Outputs

WS

Enable or disable window scaling.

SACK

Enable or disable Selective ACKnowledgement (SACK).

WSVal

Factor used to calculate the new window size.

This argument is needed only when window scaling is enabled.

nagle

Enable or disable the Nagle algorithm on TCP connections.

ackOnPush

Send immediate positive acknowledgement (ACK) on receipt of TCP packets when doing Web 2.0 PUSH.

mss

Maximum Segment Size(MSS) to use for TCP Connection(0 forces use of global setting)

maxBurst

Maximum number of TCP segments allowed in a burst.

initialCwnd

Initial maximum upper limit on the number of TCP packets that can be outstanding on the TCP link to the server.

delayedAck

Timeout for TCP delayed ACK, in milliseconds.

oooQSize

Maximum size of out-of-order packets queue. A value of 0 means no limit.

maxPktPerMss

Maximum packet per MSS value

pktPerRetx

Maximum limit on the number of packets that should be retransmitted on receiving a partial ACK.

minRTO

TCP minimum RTO (in millisec)

slowStartIncr

TCP slowstart increment factor

bufferSize

TCP Buffer size

flavor

TCP algorithm

refCnt

Number of entities using this profile

synCookie

Enable or disable the SYNCOOKIE mechanism for TCP handshake with clients. Disabling SYNCOOKIE prevents SYN attack protection on the NetScaler appliance.

KAprobeUpdateLastactivity

Update last activity for the connection after receiving keep-alive (KA) probes.

dynamicReceiveBuffering

Enable or disable dynamic receive buffering. When enabled, allows the receive buffer to be adjusted dynamically based on memory and network conditions.

Note: The buffer size argument must be set for dynamic adjustments to take place.

KA

Send periodic TCP keep-alive (KA) probes to check if peer is still up.

KAcnnIdleTime

Duration, in seconds, for the connection to be idle, before sending a keep-alive (KA) probe.

KAmxProbes

Number of keep-alive (KA) probes to be sent when not acknowledged, before assuming the peer to be down.

KAprobeInterval

Time interval, in seconds, before the next keep-alive (KA) probe, if the peer does not respond.

sendBuffsize

TCP Send Buffer size

mptcp

Enable/Disable Multi-Path TCP

EstablishClientConn

Allocating Client Conn on

tcpSegOffload

TCP Segmentation Offload

rstWindowAttenuate

RST Window Attenuation

rstMaxAck

accept RST with max ACK

TimeStamp

TCP Timestamp Option

spoofSynDrop

drop invalid SYN packets

ecn

Explicit Congestion Notification

mptcpDropDataOnPreEstSF

Enable or disable dropping data on pre established subflow.

mptcpFastOpen

Enable or disable MPTCP fastopen.

mptcpSessionTimeout

MPTCP session timeout.

dsack

Enable or disable DSACK.

ackAggregation

Enable or disable ACK Aggregation.

frto

Enable or disable FRT0 (Forward RTO-Recovery).

stateflag

State flag

devno

count

Example

```
show tcp profile [profile name]
```

ns tcpbufParam

The following operations can be performed on "ns tcpbufParam":

[set](#) | [unset](#) | [show](#)

set ns tcpbufParam

Sets the attributes for the TCP buffering per connection.

Synopsys

```
set ns tcpbufParam [-size <KBytes>] [-memLimit <MBytes>]
```

Arguments

size

TCP buffering size per connection, in kilobytes.

Default value: 64

Minimum value: 4

Maximum value: 20480

memLimit

Maximum memory, in megabytes, that can be used for buffering.

Default value: 64

unset ns tcpbufParam

Use this command to remove ns tcpbufParam settings. Refer to the set ns tcpbufParam command for meanings of the arguments.

Synopsys

```
unset ns tcpbufParam [-size] [-memLimit]
```

show ns tcpbufParam

Displays the TCP buffering configuration on the appliance.

Synopsys

```
show ns tcpbufParam
```

Outputs

size

TCP buffering size per connection, in kilobytes.

memLimit

Maximum memory, in megabytes, that can be used for buffering.

Example

An example of this command's output is as follows: TCP buffer size: 64KBytes TCP buffer p

ns timeout

The following operations can be performed on "ns timeout":

[set](#) | [unset](#) | [show](#)

set ns timeout

Sets timeout values for various aspects of the NetScaler appliance. Caution: Modifying these values can affect system performance.

Synopsys

```
set ns timeout [-zombie <positive_integer>] [-httpClient <positive_integer>] [-httpServer <positive_integer>] [-tcpClient <positive_integer>] [-tcpServer <positive_integer>] [-anyClient <positive_integer>] [-anyServer <positive_integer>] [-halfclose <positive_integer>] [-nontcpZombie <positive_integer>] [-ReducedFinTimeOut <positive_integer>] [-ReducedRstTimeOut <positive_integer>] [-NewConnIdleTimeOut <positive_integer>]
```

Arguments

zombie

Interval, in seconds, at which the NetScaler zombie cleanup process must run. This process cleans up inactive TCP connections.

Default value: 120

Minimum value: 1

Maximum value: 600

httpClient

Global idle timeout, in seconds, for client connections of HTTP service type. This value is over ridden by the client timeout that is configured on individual entities.

Minimum value: 0

Maximum value: 18000

httpServer

Global idle timeout, in seconds, for server connections of HTTP service type. This value is over ridden by the server timeout that is configured on individual entities.

Minimum value: 0

Maximum value: 18000

tcpClient

Global idle timeout, in seconds, for non-HTTP client connections of TCP service type. This value is over ridden by the client timeout that is configured on individual entities.

Minimum value: 0

Maximum value: 18000

tcpServer

Global idle timeout, in seconds, for non-HTTP server connections of TCP service type. This value is over ridden by the server timeout that is configured on entities.

Minimum value: 0

Maximum value: 18000

anyClient

Global idle timeout, in seconds, for non-TCP client connections. This value is over ridden by the client timeout that is configured on individual entities.

Minimum value: 0

Maximum value: 31536000

anyServer

Global idle timeout, in seconds, for non TCP server connections. This value is over ridden by the server timeout that is configured on individual entities.

Minimum value: 0

Maximum value: 31536000

halfclose

Idle timeout, in seconds, for connections that are in TCP half-closed state.

Default value: 10

Minimum value: 1

Maximum value: 600

nontcpZombie

Interval at which the zombie clean-up process for non-TCP connections should run. Inactive IP NAT connections will be cleaned up.

Default value: 60

Minimum value: 1

Maximum value: 600

ReducedFinTimeOut

Alternative idle timeout for new TCP NATPCB connections.

Default value: 30

Minimum value: 1

Maximum value: 300

ReducedRstTimeOut

Timer interval(in seconds) for NATPCB for tcp flow

Default value: 0

Minimum value: 0

Maximum value: 300

NewConnIdleTimeOut

Timer interval(in seconds) for new NATPCB for tcp connections.

Default value: 4

Minimum value: 1

Maximum value: 120

Example

```
set ns timeout -zombie 200
```

unset ns timeout

Use this command to remove ns timeout settings. Refer to the set ns timeout command for meanings of the arguments.

Synopsys

unset ns timeout [-zombie] [-httpClient] [-httpServer] [-tcpClient] [-tcpServer] [-anyClient] [-anyServer] [-halfclose] [-nontcpZombie] [-ReducedFinTimeOut] [-ReducedRstTimeOut] [-NewConnIdleTimeOut]

show ns timeout

Displays the timeouts configured for various NetScaler entities. Note: The timeouts having default values are not displayed.

Synopsys

show ns timeout

Outputs

zombie

Timer interval(in seconds) for zombie process that cleanup inactive TCP connections

Minimum value: 1

Maximum value: 600

Default value: 120

client

Client idle timeout (in seconds). If zero, the service-type default value is taken when service is created.

server

Server idle timeout (in seconds). If zero, the service-type default is taken when service is created.

httpClient

HTTP client idle timeout (in seconds)

Minimum value: 0

Maximum value: 18000

httpServer

HTTP server idle timeout (in seconds)

Minimum value: 0

Maximum value: 18000

tcpClient

TCP client idle timeout (in seconds)

Minimum value: 0

Maximum value: 18000

tcpServer

TCP server idle timeout (in seconds)

Minimum value: 0

Maximum value: 18000

anyClient

ANY client idle timeout (in seconds)

Minimum value: 0

Maximum value: 31536000

anyServer

ANY server idle timeout (in seconds)

Minimum value: 0

Maximum value: 31536000

halfclose

Half-closed connection timeout (in seconds)

Minimum value: 1

Maximum value: 600

Default value: 10

nontcpZombie

Timer interval(in seconds) for zombie process that cleanup inactive IP NAT connections

Minimum value: 1

Maximum value: 600

Default value: 60

ReducedFinTimeOut

Timer interval(in seconds) for NATPCB for tcp flow

ReducedRstTimeOut

Timer interval(in seconds) for NATPCB for tcp flow

NewConnIdleTimeOut

Timer interval(in seconds) for new NATPCB for tcp connections

Example

```
show ns timeout
```

ns timer

The following operations can be performed on "ns timer":

add | **rm** | **set** | **unset** | **bind** | **unbind** | **show** | **rename**

add ns timer

Create a Timer.

Synopsys

```
add ns timer <name> (-interval <integer> [<unit>]) [-comment <string>]
```

Arguments

name

Timer name.

interval

The frequency at which the policies bound to this timer are invoked. The minimum value is 20 msec. The maximum value is 20940 in seconds and 349 in minutes

Default value: 5

Minimum value: 1

Maximum value: 20940000

unit

Timer interval unit

Possible values: SEC, MIN

Default value: SEC

comment

Comments associated with this timer.

Example

```
add timer policy timer -comment "Timer that would be invoked at interval 10 sec apart."
```

rm ns timer

Remove a Timer.

Synopsys

```
rm ns timer <name>
```

Arguments

name

Timer name.

Example

```
rm ns timer timer
```

set ns timer

Set a argument values for existing timer.

Synopsys

```
set ns timer <name> [-interval <integer>] [<unit>] [-comment <string>]
```

Arguments

name

Timer name.

interval

The frequency at which the policies bound to this timer are invoked. The minimum value is 20 msec. The maximum value is 20940 in seconds and 349 in minutes

Default value: 5

Minimum value: 1

Maximum value: 20940000

unit

Timer interval unit

Possible values: SEC, MIN

Default value: SEC

comment

Comments associated with this timer.

Example

```
set ns timer timer -comment "Timer that would be invoked at interval 20 sec apart."
```

unset ns timer

Unset comment for existing timer..Refer to the set ns timer command for meanings of the arguments.

Synopsys

```
unset ns timer <name> [-interval <integer>] [<unit>] [-comment <string>]
```

Example

```
unset ns timer timer -comment
```

bind ns timer

Defines the binding relation among timer, and timer policy.

Synopsys

```
bind ns timer <name> -policyName <string> -priority <positive_integer> [-gotoPriorityExpression <expression>] [-vServer <string>] [-sampleSize <positive_integer>] [-threshold <positive_integer>]
```

Arguments

name

Timer name.

policyName

The timer policy associated with the timer.

priority

Priority with which the policy is to be bound.

Minimum value: 1

Maximum value: 2147483647

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

o If gotoPriorityExpression is not present or if it is equal to END then the policy bank evaluation ends here

o Else if the gotoPriorityExpression is equal to NEXT then the next policy in the priority order is evaluated.

o Else gotoPriorityExpression is evaluated. The result of gotoPriorityExpression (which has to be a number) is processed as follows:

- An UNDEF event is triggered if

. gotoPriorityExpression cannot be evaluated

. gotoPriorityExpression evaluates to number which is smaller than the maximum priority in the policy bank but is not same as any policy's priority

. gotoPriorityExpression evaluates to a priority that is smaller than the current policy's priority

- If the gotoPriorityExpression evaluates to the priority of the current policy then the next policy in the priority order is evaluated.

- If the gotoPriorityExpression evaluates to the priority of a policy further ahead in the list then that policy will be evaluated next.

vServer

Name of the vserver which provides the context for the rule in timer policy. When not specified it is treated as a Global Default context.

sampleSize

Denotes the sample size. Sample size value of 'x' means that previous '(x - 1)' policy's rule evaluation results and the current evaluation result are present with the binding. For example, sample size of 10 means that there is a state of previous 9 policy evaluation results and also the current policy evaluation result.

Default value: 3

Minimum value: 1

Maximum value: 32

threshold

Denotes the threshold. If the rule of the policy in the binding relation evaluates 'threshold size' number of times in 'sample size' to true, then the corresponding action is taken. Its value needs to be less than or equal to the sample size value.

Default value: 3

Minimum value: 1

Maximum value: 32

Example

```
i) bind ns timer timer_trigger -policyName timer_pol -priority 1 ii) bind ns timer time:
```

unbind ns timer

Unbind entities from timer

Synopsys

```
unbind ns timer <name> -policyName <string>
```

Arguments

name

Timer name.

policyName

The timer policy associated with the timer.

Example

```
unbind ns timer timer -policyName timer_pol
```

show ns timer

Display the Timer entities.

Synopsys

```
show ns timer [<name>]
```

Arguments

name

Timer name.

Outputs

interval

The frequency at which the policies bound to this timer are invoked. The minimum value is 20 msec. The maximum value is 20940 in seconds and 349 in minutes

unit

Timer interval unit

comment

Comments associated with this timer.

policyName

The timer policy associated with the timer.

priority

Specifies the priority of the timer policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

vServer

Name of the vserver which provides the context for the rule in timer policy. When not specified it is treated as a Global Default context.

sampleSize

Denotes the sample size. Sample size value of 'x' means that previous '(x - 1)' policy's rule evaluation results and the current evaluation result are present with the binding. For example, sample size of 10 means that there is a state of previous 9 policy evaluation results and also the current policy evaluation result.

threshold

Denotes the threshold. If the rule of the policy in the binding relation evaluates 'threshold size' number of times in 'sample size' to true, then the corresponding action is taken. Its value needs to be less than or equal to the sample size value.

stateflag

bindPolicyType

devno

count

rename ns timer

Rename a timer.

Synopsys

```
rename ns timer <name>@ <newName>@
```

Arguments

name

The name of the timer.

newName

The new name of the timer.

Example

```
rename ns timer oldname newname
```

ns trafficDomain

The following operations can be performed on "ns trafficDomain":

add | **rm** | **clear** | **bind** | **unbind** | **enable** | **disable** | **show** | **stat**

add ns trafficDomain

Configure Traffic Domain on the system.

Synopsys

```
add ns trafficDomain <td> [-aliasName <string>] [-vmac ( ENABLED | DISABLED )]
```

Arguments

td

Integer value that uniquely identifies a traffic domain.

Minimum value: 1

Maximum value: 4094

aliasName

Name of traffic domain being added.

vmac

Associate the traffic domain with a VMAC address instead of with VLANs. The NetScaler ADC then sends the VMAC address of the traffic domain in all responses to ARP queries for network entities in that domain. As a result, the ADC can segregate subsequent incoming traffic for this traffic domain on the basis of the destination MAC address, because the destination MAC address is the VMAC address of the traffic domain. After creating entities on a traffic domain, you can easily manage and monitor them by performing traffic domain level operations.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
add ns trafficDomain 1 -aliasName tdl
```

rm ns trafficDomain

Remove Traffic Domain configured.

Synopsys

```
rm ns trafficDomain <td>
```

Arguments

td

Integer value that uniquely identifies a traffic domain.

Minimum value: 1

Maximum value: 4094

Example

```
rm ns trafficDomain 1
```

clear ns trafficDomain

Remove Traffic Domain configuration.

Synopsys

```
clear ns trafficDomain <td>
```

Arguments

td

Integer value that uniquely identifies a traffic domain.

Minimum value: 1

Maximum value: 4094

bind ns trafficDomain

bind vlan or bridgegroup entities with traffic domain.

Synopsys

```
bind ns trafficDomain <td> [-vlan <positive_integer>] [-bridgegroup <positive_integer>] [-vxlan <positive_integer>]
```

Arguments

td

Integer value that uniquely identifies a traffic domain.

Minimum value: 1

Maximum value: 4094

vlan

ID of the VLAN to bind to this traffic domain. More than one VLAN can be bound to a traffic domain, but the same VLAN cannot be a part of multiple traffic domains.

Minimum value: 1

Maximum value: 4094

bridgegroup

ID of the configured bridge to bind to this traffic domain. More than one bridge group can be bound to a traffic domain, but the same bridge group cannot be a part of multiple traffic domains.

Minimum value: 1

Maximum value: 1000

vxlan

ID of the VXLAN to bind to this traffic domain. More than one VXLAN can be bound to a traffic domain, but the same VXLAN cannot be a part of multiple traffic domains.

Minimum value: 1

Maximum value: 16777215

Example


```
bind ns trafficDomain 1 -vlan 2
```

unbind ns trafficDomain

Unbind vlan or bridgegroup entities from traffic domain

Synopsys

```
unbind ns trafficDomain <td> [-vlan <positive_integer>] [-bridgegroup <positive_integer>] [-vxlan <positive_integer>]
```

Arguments

td

Integer value that uniquely identifies a traffic domain.

Minimum value: 1

Maximum value: 4094

vlan

ID of the VLAN to bind to this traffic domain. More than one VLAN can be bound to a traffic domain, but the same VLAN cannot be a part of multiple traffic domains.

Minimum value: 1

Maximum value: 4094

bridgegroup

ID of the configured bridge to bind to this traffic domain. More than one bridge group can be bound to a traffic domain, but the same bridge group cannot be a part of multiple traffic domains.

Minimum value: 1

Maximum value: 1000

vxlan

ID of the VXLAN to bind to this traffic domain. More than one VXLAN can be bound to a traffic domain, but the same VXLAN cannot be a part of multiple traffic domains.

Minimum value: 1

Maximum value: 16777215

Example

```
unbind ns trafficDomain 1 -vlan 2
```

enable ns trafficDomain

Enable TrafficDomain.

Synopsys

```
enable ns trafficDomain <td>
```

Arguments

td

Integer value that uniquely identifies a traffic domain.

Minimum value: 1

Maximum value: 4094

Example

```
enable ns trafficdomain 1
```

disable ns trafficDomain

Disable TrafficDomain.

Synopsys

```
disable ns trafficDomain <td>
```

Arguments

td

Integer value that uniquely identifies a traffic domain.

Minimum value: 1

Maximum value: 4094

Example

```
disable ns trafficdomain 1
```

show ns trafficDomain

Display Traffic Domain configuration.

Synopsys

```
show ns trafficDomain [<td>]
```

Arguments

td

Integer value that uniquely identifies a traffic domain.

Minimum value: 1

Maximum value: 4094

Outputs

aliasName

Name of traffic domain being added.

vmac

Associate the traffic domain with a VMAC address instead of with VLANs. The NetScaler ADC then sends the VMAC address of the traffic domain in all responses to ARP queries for network entities in that domain. As a result, the ADC can segregate subsequent incoming traffic for this traffic domain on the basis of the destination MAC address, because the destination MAC address is the VMAC address of the traffic domain. After creating entities on a traffic domain, you can easily manage and monitor them by performing traffic domain level operations.

stateflag

Used internally for display.

vlan

ID of the VLAN to bind to this traffic domain. More than one VLAN can be bound to a traffic domain, but the same VLAN cannot be a part of multiple traffic domains.

vxlan

ID of the VXLAN to bind to this traffic domain. More than one VXLAN can be bound to a traffic domain, but the same VXLAN cannot be a part of multiple traffic domains.

bridgegroup

ID of the configured bridge to bind to this traffic domain. More than one bridge group can be bound to a traffic domain, but the same bridge group cannot be a part of multiple traffic domains.

state

The state of TrafficDomain.

devno

count

Example

An example of the output of the show trafficDomain command is as follows: 1) TrafficDomain

stat ns trafficDomain

Display statistics for Traffic Domains(s).

Synopsys

```
stat ns trafficDomain [<td>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full )]
```

Arguments

td

An integer specifying the Traffic Domain ID. Possible values: 1 through 4094.

Minimum value: 1

Maximum value: 4094

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Packets received (TdRxPkts)

Packets received on this TD.

Packets sent (TdTxPkts)

Packets transmitted from this TD.

Packets dropped (TdDropPkts)

Inbound packets dropped on this TD by reception.

Example

```
stat ns trafficdomain 1
```

ns variable

The following operations can be performed on "ns variable":

[add](#) | [rm](#) | [show](#)

add ns variable

Create a variable for use in assignments and default syntax expressions.

Synopsys

```
add ns variable <name> -type <string> [-scope global] [-ifFull ( undef | lru )] [-ifValueTooBig ( undef | truncate )] [-ifNoValue ( undef | init )] [-init <string>] [-expires <positive_integer>] [-comment <string>]
```

Arguments

name

Variable name. This follows the same syntax rules as other default syntax expression entity names:

It must begin with an alpha character (A-Z or a-z) or an underscore (_).

The rest of the characters must be alpha, numeric (0-9) or underscores.

It cannot be re or xp (reserved for regular and XPath expressions).

It cannot be a default syntax expression reserved word (e.g. SYS or HTTP).

It cannot be used for an existing default syntax expression object (HTTP callout, patset, dataset, stringmap, or named expression).

type

Specification of the variable type; one of the following:

ulong - singleton variable with an unsigned 64-bit value.

text(value-max-size) - singleton variable with a text string value.

map(text(key-max-size),ulong,max-entries) - map of text string keys to unsigned 64-bit values.

map(text(key-max-size),text(value-max-size),max-entries) - map of text string keys to text string values.

where

value-max-size is a positive integer that is the maximum number of bytes in a text string value.

key-max-size is a positive integer that is the maximum number of bytes in a text string key.

max-entries is a positive integer that is the maximum number of entries in a map variable.

For a global singleton text variable, value-max-size <= 64000.

For a global map with ulong values, key-max-size <= 64000.

For a global map with text values, key-max-size + value-max-size <= 64000.

max-entries is a positive integer that is the maximum number of entries in a map variable. This has a theoretical maximum of $2^{64}-1$, but in actual use will be much smaller, considering the memory available for use by the map.

Example:

map(text(10),text(20),100) specifies a map of text string keys (max size 10 bytes) to text string values (max size 20 bytes), with 100 max entries.

scope

Scope of the variable:

global - (default) one set of values visible across all Packet Engines and, in a cluster, all nodes

Possible values: global

Default value: global

ifFull

Action to perform if an assignment to a map exceeds its configured max-entries:

lru - (default) reuse the least recently used entry in the map.

undef - force the assignment to return an undefined (Undef) result to the policy executing the assignment.

Possible values: undef, lru

Default value: lru

ifValueTooBig

Action to perform if an value is assigned to a text variable that exceeds its configured max-size,

or if a key is used that exceeds its configured max-size:

truncate - (default) truncate the text string to the first max-size bytes and proceed.

undef - force the assignment or expression evaluation to return an undefined (Undef) result to the policy executing the assignment or expression.

Possible values: undef, truncate

Default value: truncate

ifNoValue

Action to perform if on a variable reference in an expression if the variable is single-valued and uninitialized

or if the variable is a map and there is no value for the specified key:

init - (default) initialize the single-value variable, or create a map entry for the key and the initial value, using the -init value or its default.

undef - force the expression evaluation to return an undefined (Undef) result to the policy executing the expression.

Possible values: undef, init

Default value: init

init

Initialization value for values in this variable. Default: 0 for ulong, NULL for text

expires

Value expiration in seconds. If the value is not referenced within the expiration period it will be deleted. 0 (the default) means no expiration.

Minimum value: 0

Maximum value: 31622400

comment

Comments associated with this variable.

Example

```
add ns variable user_privilege_map -type map(text(15),text(10),10000)
```

rm ns variable

Remove a variable and its value(s).

Synopsys

```
rm ns variable <name>
```

Arguments

name

Variable name. This follows the same syntax rules as other default syntax expression entity names:

It must begin with an alpha character (A-Z or a-z) or an underscore (_).

The rest of the characters must be alpha, numeric (0-9) or underscores.

It cannot be re or xp (reserved for regular and XPath expressions).

It cannot be a default syntax expression reserved word (e.g. SYS or HTTP).

It cannot be used for an existing default syntax expression object (HTTP callout, patset, dataset, stringmap, or named expression).

Example

```
rm ns variable user_privilege_map
```

show ns variable

Display configured variables

Synopsys

```
show ns variable [<name>]
```

Arguments

name

Variable name. This follows the same syntax rules as other default syntax expression entity names:

It must begin with an alpha character (A-Z or a-z) or an underscore (_).

The rest of the characters must be alpha, numeric (0-9) or underscores.

It cannot be re or xp (reserved for regular and XPath expressions).

It cannot be a default syntax expression reserved word (e.g. SYS or HTTP).

It cannot be used for an existing default syntax expression object (HTTP callout, patset, dataset, stringmap, or named expression).

Outputs

type

Specification of the variable type; one of the following:

ulong - singleton variable with an unsigned 64-bit value.

text(value-max-size) - singleton variable with a text string value.

map(text(key-max-size),ulong,max-entries) - map of text string keys to unsigned 64-bit values.

map(text(key-max-size),text(value-max-size),max-entries) - map of text string keys to text string values.

where

value-max-size is a positive integer that is the maximum number of bytes in a text string value.

key-max-size is a positive integer that is the maximum number of bytes in a text string key.

max-entries is a positive integer that is the maximum number of entries in a map variable.

For a global singleton text variable, value-max-size <= 64000.

For a global map with ulong values, key-max-size <= 64000.

For a global map with text values, key-max-size + value-max-size <= 64000.

max-entries is a positive integer that is the maximum number of entries in a map variable. This has a theoretical maximum of $2^{64}-1$, but in actual use will be much smaller, considering the memory available for use by the map.

Example:

map(text(10),text(20),100) specifies a map of text string keys (max size 10 bytes) to text string values (max size 20 bytes), with 100 max entries.

scope

Scope of the variable:

global - (default) one set of values visible across all Packet Engines and, in a cluster, all nodes

ifFull

Action to perform if an assignment to a map exceeds its configured max-entries:

lru - (default) reuse the least recently used entry in the map.

undef - force the assignment to return an undefined (Undef) result to the policy executing the assignment.

ifValueTooBig

Action to perform if an value is assigned to a text variable that exceeds its configured max-size,

or if a key is used that exceeds its configured max-size:

truncate - (default) truncate the text string to the first max-size bytes and proceed.

undef - force the assignment or expression evaluation to return an undefined (Undef) result to the policy executing the assignment or expression.

ifNoValue

Action to perform if on a variable reference in an expression if the variable is single-valued and uninitialized

or if the variable is a map and there is no value for the specified key:

init - (default) initialize the single-value variable, or create a map entry for the key and the initial value, using the -init value or its default.

undef - force the expression evaluation to return an undefined (Undef) result to the policy executing the expression.

expires

Value expiration in seconds. If the value is not referenced within the expiration period it will be deleted. 0 (the default) means no expiration.

init

Initialization value for values in this variable. Default: 0 for ulong, NULL for text

comment

Comments associated with this variable.

builtin

Flag to determine if the variable is built-in or not

referenceCount

The number of references to the variable in expressions and assignments.

stateflag

devno

count

ns version

The following operations can be performed on "ns version":

show ns version

Displays the version and build number of the appliance.

Synopsys

show ns version

Outputs

version

Version.

Mode

Kernel mode (KMPE/VMPE).

ns weblogparam

The following operations can be performed on "ns weblogparam":

[set](#) | [unset](#) | [show](#)

set ns weblogparam

Sets the Weblog parameters.

Synopsys

```
set ns weblogparam [-bufferSizeMB <positive_integer>] [-customReqHdrs <string> ...] [-customRspHdrs <string> ...]
```

Arguments

bufferSizeMB

Buffer size, in MB, allocated for log transaction data on the system. The maximum value is limited to the memory available on the system.

Default value: 16

Minimum value: 1

Maximum value: 4294967294LU

customReqHdrs

Name(s) of HTTP request headers whose values should be exported by the Web Logging feature.

customRspHdrs

Name(s) of HTTP response headers whose values should be exported by the Web Logging feature.

unset ns weblogparam

Use this command to remove ns weblogparam settings. Refer to the set ns weblogparam command for meanings of the arguments.

Synopsys

```
unset ns weblogparam [-bufferSizeMB] [-customReqHdrs] [-customRspHdrs]
```

show ns weblogparam

Displays the Weblog parameters.

Synopsys

```
show ns weblogparam
```

Outputs

bufferSizeMB

Buffer size in MB.

customReqHdrs

Name(s) of HTTP request headers whose values should be exported by the Web Logging feature.

customRspHdrs

Name(s) of HTTP response headers whose values should be exported by the Web Logging feature.

ns xmlns:namespace

The following operations can be performed on "ns xmlns:namespace":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add ns xmlns:namespace

Adds a mapping between an XML prefix and a namespace URI (Uniform Resource Identifier).

Synopsis

```
add ns xmlns:namespace <prefix> <namespace> [-description <string>]
```

Arguments

prefix

XML prefix.

namespace

Expanded namespace for which the XML prefix is provided.

description

Description for the prefix.

Example

```
add ns xmlns:namespace soap http://schemas.xmlsoap.org/soap/envelope/
```

rm ns xmlns:namespace

Removes the mapping between an XML prefix and a namespace URI.

Synopsis

```
rm ns xmlns:namespace <prefix>
```

Arguments

prefix

XML prefix for which the mapping must be removed.

Example

```
rm ns xmlns:namespace soap
```

set ns xmlns:namespace

Modifies the mapping between an XML prefix and a namespace URI.

Synopsis

```
set ns xmlns:namespace <prefix> [<namespace>] [-description <string>]
```

Arguments

prefix

XML prefix for which the namespace or description must be added or updated.

namespace

Expanded namespace for which the XML prefix is provided.

description

Description for the prefix.

Example

```
set ns xmlnsnamespace soap -description SOAP/1.1
```

unset ns xmlnsnamespace

Use this command to remove ns xmlnsnamespace settings. Refer to the set ns xmlnsnamespace command for meanings of the arguments.

Synopsys

```
unset ns xmlnsnamespace <prefix> [-namespace] [-description]
```

show ns xmlnsnamespace

Displays the mappings between XML prefixes to namespace URIs.

Synopsys

```
show ns xmlnsnamespace [<prefix>]
```

Arguments

prefix

Name of the prefix for which the mappings must be displayed.

Outputs

namespace

Expanded namespace for which the XML prefix is provided.

description

Description for the prefix.

devno

count

stateflag

Example

```
show ns xmlnsnamespace soap
```

reboot

The following operations can be performed on "reboot":

reboot

Restarts the NetScaler appliance. Note: * When a standalone NetScaler appliance is rebooted, the unsaved configurations (configurations performed since the last 'save ns config' command was issued) are lost. * In the high availability mode, when the primary appliance is rebooted, the secondary system takes over and becomes the primary. The unsaved configurations from the old primary are available on the new primary appliance. * In a cluster setup, this command can be executed only through the cluster IP address and it reboots only the configuration coordinator.

Synopsys

reboot [-warm]

Arguments

warm

Restarts the NetScaler software without rebooting the underlying operating system. The session terminates and you must log on to the appliance after it has restarted.

Note: This argument is required only for nCore appliances. Classic appliances ignore this argument.

shutdown

The following operations can be performed on "shutdown":

shutdown

Stops all operations and powers off the NetScaler appliance. Note: * When a standalone NetScaler appliance is shut down, the unsaved configurations (configurations performed since the last 'save ns config' command was issued) are lost. * In a high availability setup, when the primary appliance is shut down, the secondary appliance takes over and becomes the primary. The unsaved configurations from the old primary are available on the new primary appliance. * In a cluster setup, this command can be executed only through the cluster IP address and it shuts down only the configuration coordinator.

Synopsys

shutdown

NTP Commands

The entities on which you can perform NetScaler CLI operations:

- o ntp param
- o ntp server
- o ntp status
- o ntp sync

ntp param

The following operations can be performed on "ntp param":

[set](#) | [unset](#) | [show](#)

set ntp param

Modifies the values for NTP parameters on the NetScaler appliance.

Synopsys

```
set ntp param [-authentication ( YES | NO )] [-trustedkey <positive_integer> ...] [-autokeyLogsec <positive_integer>]
[-revokeLogsec <positive_integer>]
```

Arguments

authentication

Apply NTP authentication, which enables the NTP client (NetScaler) to verify that the server is in fact known and trusted.

Possible values: YES, NO

Default value: YES

trustedkey

Key identifiers that are trusted for server authentication with symmetric key cryptography in the keys file.

Minimum value: 1

Maximum value: 65534

autokeyLogsec

Autokey protocol requires the keys to be refreshed periodically. This parameter specifies the interval between regenerations of new session keys. In seconds, expressed as a power of 2.

Default value: 12

Minimum value: 0

Maximum value: 32

revokeLogsec

Interval between re-randomizations of the autokey seeds to prevent brute-force attacks on the autokey algorithms.

Default value: 16

Minimum value: 0

Maximum value: 32

unset ntp param

Use this command to remove ntp param settings. Refer to the set ntp param command for meanings of the arguments.

Synopsys

```
unset ntp param [-authentication] [-trustedkey] [-autokeyLogsec] [-revokeLogsec]
```

show ntp param

Displays information about the NTP parameters.

Synopsys

show ntp param

Outputs

authentication

Apply NTP authentication, which enables the NTP client (NetScaler) to verify that the server is in fact known and trusted.

trustedkey

Key identifiers that are trusted for server authentication with symmetric key cryptography in the keys file.

autokeyLogsec

Autokey protocol requires the keys to be refreshed periodically. This parameter specifies the interval between regenerations of new session keys. In seconds, expressed as a power of 2.

revokeLogsec

Interval between re-randomizations of the autokey seeds to prevent brute-force attacks on the autokey algorithms.

ntp server

The following operations can be performed on "ntp server":

add | **rm** | **set** | **unset** | **show**

add ntp server

Adds an NTP server to the appliance. This server can be used to synchronize the time on the appliance to the network time.

Synopsys

```
add ntp server (<serverIP> | <serverName>) [-minpoll <positive_integer>] [-maxpoll <positive_integer>] [-autokey | -key <positive_integer>]
```

Arguments

serverIP

IP address of the NTP server.

serverName

Fully qualified domain name of the NTP server.

minpoll

Minimum time after which the NTP server must poll the NTP messages. In seconds, expressed as a power of 2.

Default value: NS_NTP_MINPOLL_DEFAULT_VALUE

Minimum value: 4

Maximum value: 17

maxpoll

Maximum time after which the NTP server must poll the NTP messages. In seconds, expressed as a power of 2.

Default value: NS_NTP_MAXPOLL_DEFAULT_VALUE

Minimum value: 4

Maximum value: 17

autokey

Use the Autokey protocol for key management for this server, with the cryptographic values (for example, symmetric key, host and public certificate files, and sign key) generated by the ntp-keygen utility. To require authentication for communication with the server, you must set either the value of this parameter or the key parameter.

key

Key to use for encrypting authentication fields. All packets sent to and received from the server must include authentication fields encrypted by using this key. To require authentication for communication with the server, you must set either the value of this parameter or the autokey parameter.

Minimum value: 1

Maximum value: 65534

rm ntp server

Removes an NTP server. You can specify the server by IP address or by name.

Synopsis

```
rm ntp server (<serverIP> | <serverName>)
```

Arguments

serverIP

IP address of the NTP server to be removed.

serverName

Name of the NTP server to be removed.

set ntp server

Modifies the specified attributes of an NTP server.

Synopsis

```
set ntp server (<serverIP> | <serverName>) [-minpoll <positive_integer>] [-maxpoll <positive_integer>] [-preferredNtpServer ( YES | NO )] [-autokey | -key <positive_integer>]
```

Arguments

serverIP

IP address of the NTP server to be modified.

serverName

Name of the NTP server to be modified.

minpoll

Minimum time after which the NTP server must poll the NTP messages. In seconds, expressed as a power of 2.

Default value: NS_NTP_MINPOLL_DEFAULT_VALUE

Minimum value: 4

Maximum value: 17

maxpoll

Maximum time after which the NTP server must poll the NTP messages. In seconds, expressed as a power of 2.

Default value: NS_NTP_MAXPOLL_DEFAULT_VALUE

Minimum value: 4

Maximum value: 17

preferredNtpServer

Preferred NTP server. The NetScaler appliance chooses this NTP server for time synchronization among a set of correctly operating hosts.

Possible values: YES, NO

Default value: NO

autokey

Use the Autokey protocol for key management for this server, with the cryptographic values (for example, symmetric key, host and public certificate files, and sign key) generated by the ntp-keygen utility. To require authentication for communication with the server, you must set either the value of this parameter or the key parameter.

key

Key to use for encrypting authentication fields. All packets sent to and received from the server must include authentication fields encrypted by using this key. To require authentication for communication with the server, you must set either the value of this parameter or the autokey parameter.

Minimum value: 1

Maximum value: 65534

unset ntp server

Unset the specified attributes of an NTP server..Refer to the set ntp server command for meanings of the arguments.

Synopsis

```
unset ntp server (<serverIP> | <serverName>) [-autokey] [-minpoll] [-maxpoll] [-preferredNtpServer] [-key]
```

show ntp server

Displays information about an NTP server. You can specify the server by IP address or by name.

Synopsis

```
show ntp server [<serverIP> | <serverName>]
```

Arguments

serverIP

IP address of the NTP server about which to display information.

serverName

Name of the NTP server about which to display information.

Outputs

minpoll

Minimum poll interval of the server in secs.

maxpoll

Maximum poll interval of the server in secs.

preferredNtpServer

Preferred NTP server. The NetScaler appliance chooses this NTP server for time synchronization among a set of correctly operating hosts.

autokey

Use the Autokey protocol for key management for this server, with the cryptographic values (for example, symmetric key, host and public certificate files, and sign key) generated by the ntp-keygen utility. To require authentication for communication with the server, you must set either the value of this parameter or the key parameter.

key

Key to use for encrypting authentication fields. All packets sent to and received from the server must include authentication fields encrypted by using this key. To require authentication for communication with the server, you must set either the value of this parameter or the autokey parameter.

devno

count

stateflag

ntp status

The following operations can be performed on "ntp status":

show ntp status

Displays the NTP status on the appliance.

Synopsys

show ntp status

Outputs

response

ntp sync

The following operations can be performed on "ntp sync":

[enable](#) | [disable](#) | [show](#)

enable ntp sync

Enables NTP synchronization. When NTP synchronization is enabled, the NTP daemon is spawned for time synchronization.

Synopsys

```
enable ntp sync
```

disable ntp sync

Disables NTP synchronization.

Synopsys

```
disable ntp sync
```

show ntp sync

Displays the status of the NTP synchronization.

Synopsys

```
show ntp sync
```

Outputs

state

Show NTP status

Policy Commands

The entities on which you can perform NetScaler CLI operations:

- o policy dataset
- o policy evaluation
- o policy expression
- o policy httpCallout
- o policy map
- o policy patClass
- o policy patset
- o policy stringmap

policy dataset

The following operations can be performed on "policy dataset":

`add` | `rm` | `bind` | `unbind` | `show`

add policy dataset

Adds a policy dataset to the appliance.

Synopsis

```
add policy dataset <name> <type> [-indexType ( Auto-generated | User-defined )] [-comment <string>]
```

Arguments

name

Name of the dataset. Must not exceed 127 characters.

type

Type of value to bind to the dataset.

Possible values: ipv4, number, ipv6, ulong, double, mac

indexType

Index type.

comment

Any comments to preserve information about this dataset.

Example

```
add policy dataset ts1 -type IPV4
```

rm policy dataset

Removes a dataset from the appliance.

Synopsis

```
rm policy dataset <name>
```

Arguments

name

Name of the dataset to remove.

Example

```
rm policy dataset pat1
```

bind policy dataset

Binds a value of the specified type to the dataset. If the first value is bound by using an index label, the other bind statements to that set should also provide an index.

Synopsis

bind policy dataset <name> <value> [-index <positive_integer>]

Arguments

name

Name of the dataset to which to bind the value.

value

Value of the specified type that is associated with the dataset.

index

The Index of the value associated with set.

Minimum value: 1

Maximum value: 4294967290

Example

```
bind policy dataset ts1 192.168.20.1 -index 2
```

unbind policy dataset

Unbind string(s) from a dataset.

Synopsys

unbind policy dataset <name> <value>

Arguments

name

Name of the dataset from which to unbind the value.

value

Value to unbind from the dataset.

Example

```
unbind policy dataset pat1 bar xyz
```

show policy dataset

Display the configured dataset(s).

Synopsys

show policy dataset [<name>]

Arguments

name

Name of the dataset. Must not exceed 127 characters.

Outputs

stateflag

value

Value of the specified type that is associated with the dataset.

index

The index of the value (ipv4, ipv6, number) associated with the set.

description

Description of the set

type

Type of value to bind to the dataset.

indexType

Index type.

MaxIndex

Maximum number of values bounded to dataset. The maxindex value will not be decreased when we unbind a value from the dataset. This field is used in auto-generated indexing type.

comment

Any comments to preserve information about this dataset.

devno**count**

Example

```
show policy dataset set1
```

policy evaluation

The following operations can be performed on "policy evaluation":

show policy evaluation

Executes pixl expression or action and gives result. Result type can be zero or more of: -Bool -Num -Double -Unsigned long -String

Synopsys

show policy evaluation (-expression <expression> | -action <string>) -type <type> -input <string>

Arguments

expression

Expression string. For example: http.req.body(100).contains("this").

action

Rewrite action name. Supported rewrite action types are:

- delete
- delete_all
- delete_http_header
- insert_after
- insert_after_all
- insert_before
- insert_before_all
- insert_http_header
- replace
- replace_all

type

Indicates request or response input packet

Possible values: HTTP_REQ, HTTP_RES, TEXT

input

Text representation of input packet.

Outputs

stateflag

pitModifiedInputData

Text representation of packet after evaluating expression or rewrite action.

pitBoolResult

Result of the expression in bool format.

pitNumResult

Result of the expression in num format.

pitDoubleResult

Result of the expression in double format.

pitUlongResult

Result of the expression in unsigned long format.

pitRefResult

Result of the expression in string format.

isPitEmptyRefResult

Result of the expression is empty string.

pitOffsetResult

Offset of the resultant string.

pitOffsetResultLen

Offset length of the resultant string.

isTruncatedRefResult

Identify whether ref result is truncated result.

pitBoolEvalTime

Average evaluation time of bool type expression in nanoseconds.

pitNumEvalTime

Average evaluation time of num type expression in nanoseconds.

pitDoubleEvalTime

Average evaluation time of double type expression in nanoseconds.

pitUlongEvalTime

Average evaluation time of unsigned long type expression in nanoseconds.

pitRefEvalTime

Average evaluation time of string type expression in nanoseconds.

pitOffsetEvalTime

Average evaluation time in finding offset of the resultant string in the input. Time is in nanoseconds.

pitActionEvalTime

Average evaluation time of rewrite action in nanoseconds.

pitOperationPerformerArray

Details of the operation NS performed at various offsets during applying of rewrite action on input data. Operation can be insertion, modification or deletion.

pitOldOffsetArray

Details of the offsets in the input data at which NS either inserted or modified or deleted data during applying of rewrite action.

pitNewOffsetArray

Details of the offsets in the output data at which NS either inserted or modified or deleted data during applying of rewrite action.

pitOffsetLengthArray

Details of the lengths of the data which NS either inserted or modified or deleted during applying of rewrite action.

pitBoolErrorResult

Result of the bool type expression if any error occurs during evaluation. Result will be in string format.

pitNumErrorResult

Result of the num type expression if any error occurs during evaluation. Result will be in string format.

pitDoubleErrorResult

Result of the double type expression if any error occurs during evaluation. Result will be in string format.

pitUlongErrorResult

Result of the unsigned long type expression if any error occurs during evaluation. Result will be in string format.

pitRefErrorResult

Result of the ref type expression if any error occurs during evaluation. Result will be in string format.

pitOffsetErrorResult

Result of the expression if any error occurs in calculating offset. Result will be in string format.

pitActionErrorResult

Result of the action if any error occurs in evaluation. Result will be in string format.

devno

count

Example

Example 1: `show policy evaluation -action rw_act_1 -type http_req -input 'GET / HTTP/1.1\'`

policy expression

The following operations can be performed on "policy expression":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add policy expression

Creates a classic or default syntax named expression, which can be used in multiple policies. For example, you can create the following named expressions, ExpressionA and ExpressionB: ExpressionA: http.req.body(100).contains("A") ExpressionB: http.req.body(100).contains("B") You could then create an expression of the form: <ExpressionA || ExpressionB>

Synopsys

```
add policy expression <name> <value> [-comment <string>] [-clientSecurityMessage <string>]
```

Arguments

name

Unique name for the expression. Not case sensitive. Must begin with an ASCII letter or underscore (_) character, and must consist only of ASCII alphanumeric or underscore characters. Must not begin with 're' or 'xp' or be a word reserved for use as a default syntax expression qualifier prefix (such as HTTP) or enumeration value (such as ASCII). Must not be the name of an existing named expression, pattern set, dataset, stringmap, or HTTP callout.

value

Expression string. For example: http.req.body(100).contains("this").

comment

Any comments associated with the expression. Displayed upon viewing the policy expression.

clientSecurityMessage

Message to display if the expression fails. Allowed for classic end-point check expressions only.

rm policy expression

Removes a named policy expression. If the expression is used by a policy or filter, you must remove the policy or filter before removing the expression.

Synopsys

```
rm policy expression <name> ...
```

Arguments

name

Name of the policy expression to be removed.

set policy expression

Modifies the attributes of a named policy expression.

Synopsys

```
set policy expression <name> [<value>] [-comment <string>] [-clientSecurityMessage <string>]
```

Arguments

name

Name of the policy expression to be modified.

value

The expression string.

comment

Any comments associated with the expression. Displayed upon viewing the policy expression.

clientSecurityMessage

The client security message that will be displayed on failure of this expression. Only relevant for end point check expressions.

unset policy expression

Use this command to remove policy expression settings. Refer to the set policy expression command for meanings of the arguments.

Synopsys

```
unset policy expression <name> [-comment] [-clientSecurityMessage]
```

show policy expression

Displays information about the available named policy expressions.

Synopsys

```
show policy expression [<name> | -type ( CLASSIC | ADVANCED )]
```

Arguments

name

Name of the policy expression to display. If a name is not provided, information about all policy expressions is shown.

type

Type of expression. Can be a classic or default syntax (advanced) expression.

Possible values: CLASSIC, ADVANCED

Outputs

value

The expression string.

hits

The total number of hits.

piHits

The total number of hits.

type

The type of expression. This is for output only.

clientSecurityMessage

The client security message that will be displayed on failure of the client security check.

description

Description for the expression.

comment

Any comments associated with the expression. Displayed upon viewing the policy expression.

stateflag

flag

isDefault

A value of true is returned if it is a default policy expression.

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

devno

count

policy httpCallout

The following operations can be performed on "policy httpCallout":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add policy httpCallout

Adds a default syntax expression element that, when evaluated, sends an HTTP request to a specified service and receives an HTTP response from the service. Can be used to obtain additional information for use in evaluating policy rules and other expressions. The expression prefix SYS.HTTP_CALLOUT invokes an HTTP callout. You can construct the HTTP callout request in one of two ways: * Specify individual parts of the request by using the HTTP method, host expression, URL stem expression, and header parameters. These parts are evaluated at run time and concatenated to build the request. * Specify the entire HTTP request in a single expression.

Synopsys

```
add policy httpCallout <name> [-IPAddress <ip_addr|ipv6_addr>] [-port <port>] [-vServer <string>] [-returnType  
<returnType>] [-httpMethod ( GET | POST )] [-hostExpr <string>] [-urlStemExpr <string>] [-headers <name(value)>  
...] [-parameters <name(value)> ...] [-bodyExpr <string>] [-fullReqExpr <string>] [-scheme ( http | https )] [-resultExpr  
<string>] [-cacheForSecs <secs>] [-comment <string>]
```

Arguments

name

Name for the HTTP callout. Not case sensitive. Must begin with an ASCII letter or underscore (_) character, and must consist only of ASCII alphanumeric or underscore characters. Must not begin with 're' or 'xp' or be a word reserved for use as a default syntax expression qualifier prefix (such as HTTP) or enumeration value (such as ASCII). Must not be the name of an existing named expression, pattern set, dataset, stringmap, or HTTP callout.

IPAddress

IP Address of the server (callout agent) to which the callout is sent. Can be an IPv4 or IPv6 address.

Mutually exclusive with the Virtual Server parameter. Therefore, you cannot set the <IP Address, Port> and the Virtual Server in the same HTTP callout.

port

Server port to which the HTTP callout agent is mapped. Mutually exclusive with the Virtual Server parameter. Therefore, you cannot set the <IP Address, Port> and the Virtual Server in the same HTTP callout.

Minimum value: 1

vServer

Name of the load balancing, content switching, or cache redirection virtual server (the callout agent) to which the HTTP callout is sent. The service type of the virtual server must be HTTP. Mutually exclusive with the IP address and port parameters. Therefore, you cannot set the <IP Address, Port> and the Virtual Server in the same HTTP callout.

returnType

Type of data that the target callout agent returns in response to the callout.

Available settings function as follows:

- * TEXT - Treat the returned value as a text string.
- * NUM - Treat the returned value as a number.
- * BOOL - Treat the returned value as a Boolean value.

Note: You cannot change the return type after it is set.

Possible values: BOOL, NUM, TEXT

httpMethod

Method used in the HTTP request that this callout sends. Mutually exclusive with the full HTTP request expression.

Possible values: GET, POST

hostExpr

Default Syntax string expression to configure the Host header. Can contain a literal value (for example, 10.101.10.11) or a derived value (for example, `http.req.header("Host")`). The literal value can be an IP address or a fully qualified domain name. Mutually exclusive with the full HTTP request expression.

urlStemExpr

Default Syntax string expression for generating the URL stem. Can contain a literal string (for example, `"/mysite/index.html"`) or an expression that derives the value (for example, `http.req.url`). Mutually exclusive with the full HTTP request expression.

headers

One or more headers to insert into the HTTP request. Each header is specified as `"name(expr)"`, where `expr` is a default syntax expression that is evaluated at runtime to provide the value for the named header. You can configure a maximum of eight headers for an HTTP callout. Mutually exclusive with the full HTTP request expression.

parameters

One or more query parameters to insert into the HTTP request URL (for a GET request) or into the request body (for a POST request). Each parameter is specified as `"name(expr)"`, where `expr` is a default syntax expression that is evaluated at run time to provide the value for the named parameter (`name=value`). The parameter values are URL encoded. Mutually exclusive with the full HTTP request expression.

bodyExpr

An advanced string expression for generating the body of the request. The expression can contain a literal string or an expression that derives the value (for example, `client.ip.src`). Mutually exclusive with `-fullReqExpr`.

fullReqExpr

Exact HTTP request, in the form of a default syntax expression, which the NetScaler appliance sends to the callout agent. If you set this parameter, you must not include HTTP method, host expression, URL stem expression, headers, or parameters.

The request expression is constrained by the feature for which the callout is used. For example, an HTTP.RES expression cannot be used in a request-time policy bank or in a TCP content switching policy bank.

The NetScaler appliance does not check the validity of this request. You must manually validate the request.

scheme

Type of scheme for the callout server.

Possible values: http, https

resultExpr

Expression that extracts the callout results from the response sent by the HTTP callout agent. Must be a response based expression, that is, it must begin with `HTTP.RES`. The operations in this expression must match the return type. For example, if you configure a return type of TEXT, the result expression must be a text based expression. If the return type is NUM, the result expression (`resultExpr`) must return a numeric value, as in the following example: `http.res.body(10000).length`.

cacheForSecs

Duration, in seconds, for which the callout response is cached. The cached responses are stored in an integrated caching content group named `"calloutContentGroup"`. If no duration is configured, the callout responses will not be cached unless normal caching configuration is used to cache them. This parameter takes precedence over any normal caching configuration that would otherwise apply to these responses.

Note that the calloutContentGroup definition may not be modified or removed nor may it be used with other cache policies.

Minimum value: 1

Maximum value: 31536000

comment

Any comments to preserve information about this HTTP callout.

Example

```
add policy httpcallout h1 -IPAddress 1.1.1.1 -PORT 80
```

rm policy httpCallout

Removes an HTTP callout. You cannot remove an HTTP callout that is used in any part of policy, action, or expression.

Synopsis

```
rm policy httpCallout <name>
```

Arguments

name

Name of the HTTP callout to remove.

Example

```
rm policy httpcallout h1
```

set policy httpCallout

Modifies the attributes of an existing HTTP callout element.

Synopsis

```
set policy httpCallout <name> [-IPAddress <ip_addr|ipv6_addr>] [-port <port>] [-vServer <string>] [-returnType  
<returnType>] [-httpMethod ( GET | POST )] [-hostExpr <string>] [-urlStemExpr <string>] [-headers <name(value)>  
...] [-parameters <name(value)> ...] [-bodyExpr <string>] [-fullReqExpr <string>] [-scheme ( http | https )] [-resultExpr  
<string>] [-cacheForSecs <secs>] [-comment <string>]
```

Arguments

name

Name of the HTTP callout to configure.

IPAddress

IP Address of the server (callout agent) to which the callout is sent. Can be an IPv4 or IPv6 address.

Mutually exclusive with the Virtual Server parameter. Therefore, you cannot set the <IP Address, Port> and the Virtual Server in the same HTTP callout.

port

Server port to which the HTTP callout agent is mapped. Mutually exclusive with the Virtual Server parameter. Therefore, you cannot set the <IP Address, Port> and the Virtual Server in the same HTTP callout.

Minimum value: 1

vServer

Name of the load balancing, content switching, or cache redirection virtual server (the callout agent) to which the HTTP callout is sent. The service type of the virtual server must be HTTP. Mutually exclusive with the IP address and port parameters. Therefore, you cannot set the <IP Address, Port> and the Virtual Server in the same HTTP callout.

returnType

Type of data that the target callout agent returns in response to the callout.

Available settings function as follows:

- * TEXT - Treat the returned value as a text string.
- * NUM - Treat the returned value as a number.
- * BOOL - Treat the returned value as a Boolean value.

Note: You cannot change the return type after it is set.

Possible values: BOOL, NUM, TEXT

httpMethod

Method used in the HTTP request that this callout sends. Mutually exclusive with the full HTTP request expression.

Possible values: GET, POST

hostExpr

Default Syntax string expression to configure the Host header. Can contain a literal value (for example, 10.101.10.11) or a derived value (for example, http.req.header("Host")). The literal value can be an IP address or a fully qualified domain name. Mutually exclusive with the full HTTP request expression.

urlStemExpr

Default Syntax string expression for generating the URL stem. Can contain a literal string (for example, "/mysite/index.html") or an expression that derives the value (for example, http.req.url). Mutually exclusive with the full HTTP request expression.

headers

One or more headers to insert into the HTTP request. Each header is specified as "name(expr)", where expr is a default syntax expression that is evaluated at runtime to provide the value for the named header. You can configure a maximum of eight headers for an HTTP callout. Mutually exclusive with the full HTTP request expression.

parameters

One or more query parameters to insert into the HTTP request URL (for a GET request) or into the request body (for a POST request). Each parameter is specified as "name(expr)", where expr is an default syntax expression that is evaluated at run time to provide the value for the named parameter (name=value). The parameter values are URL encoded. Mutually exclusive with the full HTTP request expression.

bodyExpr

An advanced string expression for generating the body of the request. The expression can contain a literal string or an expression that derives the value (for example, client.ip.src). Mutually exclusive with -fullReqExpr.

fullReqExpr

Exact HTTP request, in the form of a default syntax expression, which the NetScaler appliance sends to the callout agent. If you set this parameter, you must not include HTTP method, host expression, URL stem expression, headers, or parameters.

The request expression is constrained by the feature for which the callout is used. For example, an HTTP.RES expression cannot be used in a request-time policy bank or in a TCP content switching policy bank.

The NetScaler appliance does not check the validity of this request. You must manually validate the request.

scheme

Type of scheme for the callout server.

Possible values: http, https

resultExpr

Expression that extracts the callout results from the response sent by the HTTP callout agent. Must be a response based expression, that is, it must begin with HTTP.RES. The operations in this expression must match the return type. For example, if you configure a return type of TEXT, the result expression must be a text based expression. If the return type is NUM, the result expression (resultExpr) must return a numeric value, as in the following example: http.res.body(10000).length.

cacheForSecs

Duration, in seconds, for which the callout response is cached. The cached responses are stored in an integrated caching content group named "calloutContentGroup". If no duration is configured, the callout responses will not be cached unless normal caching configuration is used to cache them. This parameter takes precedence over any normal caching configuration that would otherwise apply to these responses.

Note that the calloutContentGroup definition may not be modified or removed nor may it be used with other cache policies.

Minimum value: 1

Maximum value: 31536000

comment

Any comments to preserve information about this HTTP callout.

Example

```
set policy httpcallout h1 -IPAddress 1.1.1.1 -PORT 80
```

unset policy httpCallout

Use this command to remove policy httpCallout settings. Refer to the set policy httpCallout command for meanings of the arguments.

Synopsys

```
unset policy httpCallout <name> [-IPAddress] [-port] [-vServer] [-httpMethod] [-hostExpr] [-urlStemExpr] [-headers] [-parameters] [-bodyExpr] [-fullReqExpr] [-resultExpr] [-cacheForSecs] [-comment]
```

show policy httpCallout

Displays information about the configured HTTP callouts.

Synopsys

```
show policy httpCallout [<name>]
```

Arguments

name

Name of the HTTP callout to display. If a name is not provided, information about all configured HTTP callouts is shown.

Outputs

stateflag

IPAddress

Server IP address.

port

Server port.

vServer

Vserver name

returnType

Return type of the http callout

scheme

Type of scheme(HTTP/HTTPS) for the callout server

httpMethod

Http callout request type

hostExpr

PI string expression for Host

urlStemExpr

PI string expression for URL stem

headers

PI string expression for request http headers

parameters

PI string expression for request query parameters

fullReqExpr

PI string expression for full http callout request

resultExpr

PI string expression for http callout response

hits

Total hits

undefHits

Total undefs

svrState

The state of the service

undefReason

Reason for last undef

recursiveCallout

Number of recursive callouts

bodyExpr

An advanced string expression for generating the body of the request. The expression can contain a literal string or an expression that derives the value (for example, client.ip.src). Mutually exclusive with -fullReqExpr.

cacheForSecs

Duration, in seconds, for which the callout response is cached. The cached responses are stored in an integrated caching content group named "calloutContentGroup". If no duration is configured, the callout responses will not be cached unless normal caching configuration is used to cache them. This parameter takes precedence over any normal caching configuration that would otherwise apply to these responses.

Note that the calloutContentGroup definition may not be modified or removed nor may it be used with other cache policies.

comment

Any comments to preserve information about this HTTP callout.

devno**count**

Example

```
show policy httpcallout h1
```

policy map

The following operations can be performed on "policy map":

[add](#) | [rm](#) | [show](#)

add policy map

Creates a policy to map a publicly known domain name to a target domain name for a reverse proxy virtual server used by the cache redirection feature. Optionally, you can also specify a source and target URL. The map policy can be associated with a reverse proxy cache redirection virtual server by using the 'bind cr vserver' command. There can be only one default map policy for a domain.

Synopsys

```
add policy map <mapPolicyName> -sd <string> [-su <string>] [-td <string>] [-tu <string>]
```

Arguments

mapPolicyName

Name for the map policy. Must begin with a letter, number, or the underscore (_) character and must consist only of letters, numbers, and the hash (#), period (.), colon (:), space (), at (@), equals (=), hyphen (-), and underscore (_) characters.

CLI Users: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my map" or 'my map').

sd

Publicly known source domain name. This is the domain name with which a client request arrives at a reverse proxy virtual server for cache redirection. If you specify a source domain, you must specify a target domain.

su

Source URL. Specify all or part of the source URL, in the following format: /[[prefix] [*]] [.suffix].

td

Target domain name sent to the server. The source domain name is replaced with this domain name.

tu

Target URL. Specify the target URL in the following format: /[[prefix] [*]][.suffix].

Example

Example 1 The following example creates a default map policy (map1) for the source domain

rm policy map

Removes a map policy. Before removing the map policy, you must unbind the map policy from the reverse proxy virtual server.

Synopsys

```
rm policy map <mapPolicyName>
```

Arguments

mapPolicyName

Name of the policy map to remove.

show policy map

Displays information about the available policy maps.

Synopsys

show policy map [<mapPolicyName>]

Arguments

mapPolicyName

Name of the policy map to display. If a name is not provided, information of all configured policy maps is shown.

Outputs

sd

Publicly known source domain name. This is the domain name with which a client request arrives at a reverse proxy virtual server for cache redirection. If you specify a source domain, you must specify a target domain.

su

The source URL.

td

The domain name sent to the server.

tu

The target URL.

targetName

The expression string.

devno

count

stateflag

policy patClass

The following operations can be performed on "policy patClass":

add | **rm** | **bind** | **unbind** | **show**

add policy patClass

Adds a pattern class. Each pattern class is identified by a name. More patterns (strings) can be associated with it later.
NOTE: This command is deprecated. This command is deprecated in favor of 'add policy patset'. NOTE: This command is deprecated. This command is deprecated in favor of 'add policy patset'

Synopsys

Arguments

name

Name of the pattern class. The name must not exceed 127 characters.

string

String associated with the patclass.

Example

```
add policy patclass pat1 foo
```

rm policy patClass

Removes a pattern class. Once the pattern class is removed, all the expressions referring it have undefined values.
NOTE: This command is deprecated in favor of 'rm policy patset'. NOTE: This command is deprecated. This command is deprecated in favor of 'rm policy patset'

Synopsys

Arguments

name

Name of the pattern class.

Example

```
rm policy patclass pat1
```

bind policy patClass

Binds string(s) to a pattern class. NOTE: This command is deprecated in favor of 'bind policy patset'. NOTE: This command is deprecated. This command is deprecated in favor of 'bind policy patset'

Synopsys

Arguments

name

Name of the pattern class.

string

String associated with the patclass.

Example

```
bind policy patclass pat1 bar xyz
```

unbind policy patClass

Unbinds string(s) from a pattern class. NOTE: This command is deprecated in favor of 'unbind policy patset'. NOTE: This command is deprecated. This command is deprecated in favor of 'unbind policy patset'

Synopsys

Arguments

name

Name of the pattern class.

string

String associated with the patclass.

Example

```
unbind policy patclass pat1 bar xyz
```

show policy patClass

Displays the configured pattern class(es). NOTE: This command is deprecated in favor of 'show policy patset'. NOTE: This command is deprecated. This command is deprecated in favor of 'show policy patset'

Synopsys

Arguments

name

Name of the pattern class.

Outputs

stateflag

string

String associated with the patclass.

index

The index of the string associated with the patclass.

charset

The character set of the string associated with patset.

description

Description of the patclass

isDefault

devno

count

Example

```
show policy patchclass pat1
```

policy patset

The following operations can be performed on "policy patset":

`add | rm | bind | unbind | show`

add policy patset

Adds a pattern set. A pattern set contains a name and one or more string patterns. Pattern sets can be used in default syntax expressions to match a set of strings. For example, `HTTP.REQ.URL.EQUALS_ANY("test_urls")`, where `test_urls` is a pattern set containing URL strings. Pattern sets can also be used in the search parameter of a rewrite action. Each string pattern is assigned an index that enables you to select the associated string from the set.

Synopsys

```
add policy patset <name> [-indexType ( Auto-generated | User-defined )] [-comment <string>]
```

Arguments

name

Unique name of the pattern set. Not case sensitive. Must begin with an ASCII letter or underscore (_) character and must contain only alphanumeric and underscore characters. Must not be the name of an existing named expression, pattern set, dataset, string map, or HTTP callout.

indexType

Index type.

comment

Any comments to preserve information about this patset.

Example

```
add policy patset pat1
```

rm policy patset

Removes a pattern set. If the pattern set is used by an expression in another object, such as a policy, you must remove the object before removing the pattern set.

Synopsys

```
rm policy patset <name>
```

Arguments

name

Name of the pattern set to remove.

Example

```
rm policy patset pat1
```

bind policy patset

Binds a string to a pattern set.

Synopsys

bind policy patset <name> <string> [-index <positive_integer>] [-charset (ASCII | UTF_8)]

Arguments

name

Name of the pattern set to which to bind the string.

string

String of characters that constitutes a pattern. For more information about the characters that can be used, refer to the character set parameter.

Note: Minimum length for pattern sets used in rewrite actions of type REPLACE_ALL, DELETE_ALL, INSERT_AFTER_ALL, and INSERT_BEFORE_ALL, is three characters.

index

Integer that identifies the string pattern within the pattern set. You can assign index values or allow them to be assigned automatically. If you specify an index for the first pattern that you bind to the set, you must do so for each subsequent pattern. If you do not specify an index for the first pattern, the NetScaler generates an index. If you subsequently specify an index when binding a pattern to the set, an error message appears.

The pattern index of a matching pattern can be used within default syntax expressions. For example, HTTP.REQ.URL.EQUALS_INDEX("test_url").EQ(5), returns true if the request URL matches the strings in the test_url pattern set with index 5.

Minimum value: 1

Maximum value: 4294967290

charset

Character set associated with the characters in the string.

Note: UTF-8 characters can be entered directly (if the UI supports it) or can be encoded as a sequence of hexadecimal bytes '\xNN'. For example, the UTF-8 character '?' can be encoded as '\xC3\xBC'.

Possible values: ASCII, UTF_8

Example

```
bind policy patset pat1 bar -index 2
```

unbind policy patset

Unbinds a string from a pattern set.

Synopsis

```
unbind policy patset <name> <string> ...
```

Arguments

name

Name of the pattern set from which to unbind a string.

string

String of characters to unbind from the pattern set.

Example

```
unbind policy patset pat1 bar xyz
```

show policy patset

Displays the list of pattern sets configured on the appliance.

Synopsys

show policy patset [<name>]

Arguments

name

Name of the pattern set for which to display the detailed information. If a name is not provided, a list of all pattern sets configured on the appliance is shown.

Outputs

stateflag

string

String of characters that constitutes a pattern. For more information about the characters that can be used, refer to the character set parameter.

Note: Minimum length for pattern sets used in rewrite actions of type REPLACE_ALL, DELETE_ALL, INSERT_AFTER_ALL, and INSERT_BEFORE_ALL, is three characters.

index

The index of the string associated with the patset.

charset

Character set associated with the characters in the string.

Note: UTF-8 characters can be entered directly (if the UI supports it) or can be encoded as a sequence of hexadecimal bytes '\xNN'. For example, the UTF-8 character '?' can be encoded as '\xC3\xBC'.

description

Description of the patset

isDefault

indexType

Index type.

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

MaxIndex

Maximum number of patterns bounded to pattern set. The maxindex value will not be decreased when we unbind a pattern from the patset. This field is used in auto-generated indexing type.

comment

Any comments to preserve information about this patset.

devno

count

Example

```
show policy patset pat1
```


policy stringmap

The following operations can be performed on "policy stringmap":

add | **rm** | **set** | **unset** | **bind** | **unbind** | **show**

add policy stringmap

Creates a string map. You must use the 'bind policy stringmap' command to bind strings to this string map.

Synopsis

add policy stringmap <name> [-comment <string>]

Arguments

name

Unique name for the string map. Not case sensitive. Must begin with an ASCII letter or underscore (_) character, and must consist only of ASCII alphanumeric or underscore characters. Must not begin with 're' or 'xp' or be a word reserved for use as a default syntax expression qualifier prefix (such as HTTP) or enumeration value (such as ASCII). Must not be the name of an existing named expression, pattern set, dataset, string map, or HTTP callout.

comment

Comments associated with the string map.

Example

```
i) add stringmap custom_stringmap . This creates a new string map with name custom_stringmap
```

rm policy stringmap

Removes a string map. String maps can be removed only if not used in any part of policy, action, or expression.

Synopsis

rm policy stringmap <name>

Arguments

name

Name of the string map to remove.

Example

```
i) rm stringmap custom_stringmap . This removes a string map whose name is custom_stringmap
```

set policy stringmap

Modifies the attributes of an existing string map.

Synopsis

set policy stringmap <name> -comment <string>

Arguments

name

Name of the string map to be modified.

comment

Comments associated with the string map.

Example

```
i) set stringmap custom_stringmap -comment "custom string map is for URLs." . This update;
```

unset policy stringmap

Use this command to remove policy stringmap settings. Refer to the set policy stringmap command for meanings of the arguments.

Synopsis

```
unset policy stringmap <name> -comment
```

bind policy stringmap

Binds a key and its associated value to a string map. If the key already exists and has a different value, the old value is overwritten with the new value.

Synopsis

```
bind policy stringmap <name> <key> <value>
```

Arguments

name

Name of the string map to which to bind the key-value pair.

key

Character string constituting the key to be bound to the string map. The key is matched against the data processed by the operation that uses the string map. The default character set is ASCII. UTF-8 characters can be included if the character set is UTF-8. UTF-8 characters can be entered directly (if the UI supports it) or can be encoded as a sequence of hexadecimal bytes '\xNN'. For example, the UTF-8 character '?' can be encoded as '\xC3\xBC'.

value

Character string constituting the value associated with the key. This value is returned when processed data matches the associated key. Refer to the key parameter for details of the value character set.

Example

```
bind stringmap custom_stringmap "key-string" "value-string" . This adds the key "key-strin
```

unbind policy stringmap

Removes a key from the string map.

Synopsis

```
unbind policy stringmap <name> <key>
```

Arguments

name

Name of the string map from which to remove a key.

key

Key to remove from the string map.

Example

```
unbind stringmap custom_stringmap key1 . This removes the key "key1" and its associated v:
```

show policy stringmap

Displays a list of available string maps.

Synopsys

```
show policy stringmap [<name>]
```

Arguments

name

Name of the string map to display. If a name is not provided, a list of all the configured string maps is shown.

Outputs

stateflag**comment**

Comments associated with the string map.

key

Character string constituting the key to be bound to the string map. The key is matched against the data processed by the operation that uses the string map. The default character set is ASCII. UTF-8 characters can be included if the character set is UTF-8. UTF-8 characters can be entered directly (if the UI supports it) or can be encoded as a sequence of hexadecimal bytes '\xNN'. For example, the UTF-8 character '?' can be encoded as '\xC3\xBC'.

value

Character string constituting the value associated with the key. This value is returned when processed data matches the associated key. Refer to the key parameter for details of the value character set.

devno**count**

Example

```
show stringmap custom_stringmap . Displays all the key-value pairs of a string map whose 1
```

PQ Commands

The entities on which you can perform NetScaler CLI operations:

- `pq`
- `pq binding`
- `pq policy`
- `pq stats`

pq

The following operations can be performed on "pq":

stat pq

Displays statistics of priority queuing.

Synopsys

```
stat pq [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

Policy hits (PolMatch)

Number of times the Netscaler appliance matched an incoming request using any priority queuing policy.

Threshold failed (ThrsFail)

Number of times the Netscaler appliance failed to match an incoming request to any of priority queing policy.

Priority 1 requests (Pri1Req)

Number of priority 1 requests that the Netscaler appliance received.

Priority 2 requests (Pri2Req)

Number of priority 2 requests that the Netscaler appliance received.

Priority 3 requests (Pri3Req)

Number of priority 3 requests that the Netscaler appliance received.

pq binding

The following operations can be performed on "pq binding":

show pq binding

Displays the information about the priority queuing policy bound to the virtual server. NOTE: This command is deprecated. Deprecated as cluster don't support reverse binding

Synopsys

Arguments

vServerName

Name of the load balancing virtual server for which to display the priority queuing policy.

Outputs

stateflag

policyName

The name of the priority queuing policy.

rule

The condition for applying the policy.

priority

The priority of queuing the request.

weight

Weight.

qDepth

Queue Depth.

polqDepth

Policy Queue Depth.

hits

Total number of hits.

devno

count

pq policy

The following operations can be performed on "pq policy":

add | **rm** | **set** | **unset** | **show** | **stat**

add pq policy

Adds a priority queuing policy to the appliance. Note: To use the priority queuing policy on a virtual server, the virtual server must have priority queuing enabled and the priority queuing policy must be bound to the load balancing virtual server. To enable priority queuing on the virtual server and to bind the policy, use the `set lb vserver` and `bind lb vserver` commands.

Synopsys

```
add pq policy <policyName> -rule <expression> -priority <positive_integer> [-weight <positive_integer>] [-qDepth <positive_integer>] [-polqDepth <positive_integer>]
```

Arguments

policyName

Name for the priority queuing policy. Must begin with a letter, number, or the underscore symbol (`_`). Other characters allowed, after the first character, are the hyphen (`-`), period (`.`), hash (`#`), space (), at (`@`), equals (`=`), and colon (`:`) characters.

rule

Expression or name of a named expression, against which the request is evaluated. The priority queuing policy is applied if the rule evaluates to true.

Note:

- * On the command line interface, if the expression includes blank spaces, the entire expression must be enclosed in double quotation marks.

- * If the expression itself includes double quotation marks, you must escape the quotations by using the `\` character.

- * Alternatively, you can use single quotation marks to enclose the rule, in which case you will not have to escape the double quotation marks.

- * Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the `+` operator. For example, you can create a 500-character string as follows: `"<string of 255 characters>" + "<string of 245 characters>"`

priority

Priority for queuing the request. If server resources are not available for a request that matches the configured rule, this option specifies a priority for queuing the request until the server resources are available again. Enter the value of `positive_integer` as 1, 2 or 3. The highest priority level is 1 and the lowest priority value is 3.

Minimum value: 1

Maximum value: 3

weight

Weight of the priority. Each priority is assigned a weight according to which it is served when server resources are available. The weight for a higher priority request must be set higher than that of a lower priority request.

To prevent delays for low-priority requests across multiple priority levels, you can configure weighted queuing for serving requests. The default weights for the priorities

are:

* Gold - Priority 1 - Weight 3

* Silver - Priority 2 - Weight 2

* Bronze - Priority 3 - Weight 1

Specify the weights as 0 through 101. A weight of 0 indicates that the particular priority level should be served only when there are no requests in any of the priority queues.

A weight of 101 specifies a weight of infinity. This means that this priority level is served irrespective of the number of clients waiting in other priority queues.

Minimum value: 0

Maximum value: 101

qDepth

Queue depth threshold value. When the queue size (number of requests in the queue) on the virtual server to which this policy is bound, increases to the specified qDepth value, subsequent requests are dropped to the lowest priority level.

Default value: 0

Minimum value: 0

Maximum value: 4294967294

polqDepth

Policy queue depth threshold value. When the policy queue size (number of requests in all the queues belonging to this policy) increases to the specified polqDepth value, subsequent requests are dropped to the lowest priority level.

Default value: 0

Minimum value: 0

Maximum value: 4294967294

rm pq policy

Removes a priority queuing policy from the appliance.

Synopsis

```
rm pq policy <policyName> ...
```

Arguments

policyName

Name of the priority queuing policy to be removed.

set pq policy

Modifies the attributes of a priority queuing policy.

Synopsis

```
set pq policy <policyName> [-weight <positive_integer>] [-qDepth <positive_integer> | -polqDepth <positive_integer>]
```

Arguments

policyName

Name of the priority queuing policy to be modified.

weight

Weight of the priority. Each priority is assigned a weight according to which it is served when server resources are available. The weight for a higher priority request must be set higher than that of a lower priority request.

To prevent delays for low-priority requests across multiple priority levels, you can configure weighted queuing for serving requests. The default weights for the priorities

are:

* Gold - Priority 1 - Weight 3

* Silver - Priority 2 - Weight 2

* Bronze - Priority 3 - Weight 1

Specify the weights as 0 through 101. A weight of 0 indicates that the particular priority level should be served only when there are no requests in any of the priority queues.

A weight of 101 specifies a weight of infinity. This means that this priority level is served irrespective of the number of clients waiting in other priority queues.

Minimum value: 0

Maximum value: 101

qDepth

Queue depth threshold value. When the queue size (number of requests in the queue) on the virtual server to which this policy is bound, increases to the specified qDepth value, subsequent requests are dropped to the lowest priority level.

Default value: 0

Minimum value: 0

Maximum value: 4294967294

polqDepth

Policy queue depth threshold value. When the policy queue size (number of requests in all the queues belonging to this policy) increases to the specified polqDepth value, subsequent requests are dropped to the lowest priority level.

Default value: 0

Minimum value: 0

Maximum value: 4294967294

unset pq policy

Use this command to remove pq policy settings. Refer to the set pq policy command for meanings of the arguments.

Synopsis

```
unset pq policy <policyName> [-weight] [-qDepth] [-polqDepth]
```

show pq policy

Displays information about the priority queuing policy.

Synopsis

```
show pq policy [<policyName>]
```

Arguments

policyName

Name of the priority queuing policy about which to display information. If a name is not provided, information about all priority queuing policies is shown.

Outputs

stateflag

rule

The condition for applying the policy.

priority

The priority of queuing the request.

weight

Weight.

qDepth

Queue Depth.

polqDepth

Policy Queue Depth.

hits

Total number of hits.

devno

count

stat pq policy

Displays statistics of the priority queuing policy.

Synopsys

```
stat pq policy [<policyName>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

policyName

Name of the priority queuing policy whose statistics must be displayed. If a name is not provided, statistics of all priority queuing policies are shown.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Toatal queue wait time (QWaitTim)

Amount of time spent by priority queuing clients waiting in the priority queue.

Avg queue wait time (AvWtTime)

Average wait time for clients for this priority queuing policy.

Avg clt transaction time (AvgTime)

Average time taken by a priority queuing client to complete its transaction for this priority queuing policy.

Vserver IP (VsIP)

IP address of the virtual server to which this priority queuing policy is bound.

Vserver port (VsPort)

Port number of the virtual server to which this priority queuing policy is bound.

Current queue depth (Qdepth)

Number of clients waiting currently for this priority queuing policy.

Current server connections (ServCons)

Current number of server connections established for serving clients for this priority queuing policy.

Server TCP connections (TotServCon)

Total number of server connections established for serving clients for this priority queuing policy.

Client requests dropped (Dropped)

Total number of dropped transactions for this priority queuing policy.

Client HTTP transactions (Cltrns)

Total number of client transactions for this priority queuing policy.

Queue depth (TotQLen)

Total number of waiting clients for this priority queuing policy.

Avg clt transaction time (us) (AvgTime)

Average time taken by a priority queuing client to complete its transaction for this priority queuing policy.

pq stats

The following operations can be performed on "pq stats":

show pq stats

show pq stats is an alias for stat pq

Synopsys

show pq stats - alias for 'stat pq'

Protocol Commands

The entities on which you can perform NetScaler CLI operations:

- o protocol http
- o protocol httpBand
- o protocol icmp
- o protocol icmpv6
- o protocol ip
- o protocol ipv6
- o protocol tcp
- o protocol udp

protocol http

The following operations can be performed on "protocol http":

stat protocol http

Displays statistics of the HTTP protocol.

Synopsys

```
stat protocol http [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

Total SPDY requests (SPDYStrm)

Total number of requests received over SPDYv2 and SPDYv3

Total requests (HTReqRx)

Total number of HTTP requests received.

Total responses (HTRspRx)

Total number of HTTP responses sent.

Request bytes received (HTReqbRx)

Total number of bytes of HTTP request data received.

Response bytes received (HTRspbRx)

Total number of bytes of HTTP response data received.

GETs (HTGETs)

Total number of HTTP requests received with the GET method.

POSTs (HTPOSTs)

Total number of HTTP requests received with the POST method.

Other methods (HTOthers)

Total number of HTTP requests received with methods other than GET and POST. Some of the other well-defined HTTP methods are HEAD, PUT, DELETE, OPTIONS, and TRACE. User-defined methods are also allowed.

HTTP/1.0 requests (HT10ReqRx)

Total number of HTTP/1.0 requests received.

HTTP/1.1 requests (HT11ReqRx)

Total number of HTTP/1.1 requests received.

Content-length requests (HTCLnReq)

Total number of HTTP requests in which the Content-length field of the HTTP header has been set. Content-length specifies the length of the content, in bytes, in the associated HTTP body.

Chunked requests (HTChkReq)

Total number of HTTP requests in which the Transfer-Encoding field of the HTTP header has been set to chunked.

Request bytes transmitted (HTReqbTx)

Total number of bytes of HTTP request data transmitted.

HTTP/1.0 responses (HT10RspRx)

Total number of HTTP/1.0 responses sent.

HTTP/1.1 responses (HT11RspRx)

Total number of HTTP/1.1 responses sent.

Content-length responses (HTCLnRsp)

Total number of HTTP responses sent in which the Content-length field of the HTTP header has been set. Content-length specifies the length of the content, in bytes, in the associated HTTP body.

Chunked responses (HTChunk)

Total number of HTTP responses sent in which the Transfer-Encoding field of the HTTP header has been set to chunked. This setting is used when the server wants to start sending the response before knowing its total length. The server breaks the response into chunks and sends them in sequence, inserting the length of each chunk before the actual data. The message ends with a chunk of size zero.

Multi-part responses (HTMPrtHd)

Total number of HTTP multi-part responses sent. In multi-part responses, one or more entities are encapsulated within the body of a single message.

FIN-terminated responses (HTNoCLnChunk)

Total number of FIN-terminated responses sent. In FIN-terminated responses, the server finishes sending the data and closes the connection.

Response bytes transmitted (HTRspbTx)

Total number of bytes of HTTP response data transmitted.

Incomplete headers (HTIncHd)

Total number of HTTP requests and responses received in which the HTTP header spans more than one packet.

Incomplete request headers (HTIncReqHd)

Total number of HTTP requests received in which the header spans more than one packet.

Incomplete response headers (HTIncRspHd)

Total number of HTTP responses received in which the header spans more than one packet.

HTTP 500 Server-busy Responses (HT500Rsp)

Total number of HTTP error responses received. Some of the error responses are:

500 Internal Server Error

501 Not Implemented

502 Bad Gateway

503 Service Unavailable

504 Gateway Timeout

505 HTTP Version Not Supported

Large/Invalid messages (HTInvReq)

Total number of requests and responses received with large body.

Large/Invalid chunk requests (HTInvChkRx)

Total number of requests received with large chunk size, in which the Transfer-Encoding field of the HTTP header has been set to chunked.

Large/Invalid content-length (HTInvCLn)

Total number of requests received with large content, in which the Content-length field of the HTTP header has been set. Content-length specifies the length of the content, in bytes, in the associated HTTP body.

SPDYv2 requests (SPDY2Strm)

Total number of requests received over SPDYv2

SPDYv3 requests (SPDY3Strm)

Total number of requests received over SPDYv3

protocol httpBand

The following operations can be performed on "protocol httpBand":

[set](#) | [unset](#) | [show](#)

set protocol httpBand

Sets the band size for HTTP request/response band statistics.

Synopsys

```
set protocol httpBand [-reqBandSize <integer>] [-respBandSize <integer>]
```

Arguments

reqBandSize

Band size, in bytes, for HTTP request band statistics. For example, if you specify a band size of 100 bytes, statistics will be maintained and displayed for the following size ranges:

0 - 99 bytes

100 - 199 bytes

200 - 299 bytes and so on.

Default value: 100

Minimum value: 50

respBandSize

Band size, in bytes, for HTTP response band statistics. For example, if you specify a band size of 100 bytes, statistics will be maintained and displayed for the following size ranges:

0 - 99 bytes

100 - 199 bytes

200 - 299 bytes and so on.

Default value: 1024

Minimum value: 50

Example

```
set protocol httpBand -reqBandSize 200 -respBandSize 2048
```

unset protocol httpBand

Use this command to remove protocol httpBand settings. Refer to the set protocol httpBand command for meanings of the arguments.

Synopsys

```
unset protocol httpBand [-reqBandSize] [-respBandSize]
```

show protocol httpBand

Displays statistics of the HTTP request/response band.

Synopsys

show protocol httpBand -type (REQUEST | RESPONSE)

Arguments

type

Type of statistics to display.

Possible values: REQUEST, RESPONSE

Outputs

BandRange

The range of the HTTP request/response size, in bytes.

numberofbands

The total number of http bands.

TotalBandSize

The total size of all HTTP requests/responses in this size range.

AvgBandSize

The average size of all HTTP requests/responses in this size range.

AvgBandSizeNew

The average size of all HTTP requests/responses in this size range.

BandData

The total size of all HTTP requests/responses in this size range, expressed as a percentage of the total size of all HTTP requests/responses.

BandDataNew

The total size of all HTTP requests/responses in this size range, expressed as a percentage of the total size of all HTTP requests/responses.

AccessCount

The number of HTTP requests/responses in this size range.

AccessRatio

The number of HTTP requests/responses in this size range, expressed as a percentage of the total number of HTTP requests/responses.

AccessRatioNew

The number of HTTP requests/responses in this size range, expressed as a percentage of the total number of HTTP requests/responses.

Totals

The total of totalBandSize, avgBandSize, BandData, accessCount, accessRatio respectively.

protocol icmp

The following operations can be performed on "protocol icmp":

stat protocol icmp

Displays statistics of the ICMP protocol.

Synopsys

```
stat protocol icmp [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

ICMP packets received (ICPktRx)

ICMP packets received.

ICMP bytes received (ICbRx)

Bytes of ICMP data received.

ICMP packets transmitted (ICPktTx)

ICMP packets transmitted.

ICMP bytes transmitted (ICbTx)

Bytes of ICMP data transmitted.

ICMP echo replies received (ECOREpRx)

ICMP Ping echo replies received.

ICMP echo replies transmitted (ECOREpTx)

ICMP Ping echo replies transmitted.

ICMP echos received (ECORx)

ICMP Ping Echo Request and Echo Reply packets received.

MTU lookup on dst ip info recvd (MTULkDst)

Total number of MTU lookup on destination IP info received on a need fragmentation ICMP error message failed.

ICMP rate threshold (pkts/sec) (ICThs)

Limit for ICMP packets handled every 10 milliseconds. Default value, 0, applies no limit.

This is a configurable value using the set rateControl command.

ICMP port unreachable received (PortUnRx)

ICMP Port Unreachable error messages received. This error is generated when there is no service is running on the port.

ICMP port unreachable generated (PortUnTx)

ICMP Port Unreachable error messages generated. This error is generated when there is no service is running on the port.

Need fragmentation received (NeedFrag)

ICMP Fragmentation Needed error messages received for packets that need to be fragmented but for which Don't Fragment is specified the header.

ICMP rate threshold exceeded (ICRtEx)

Times the ICMP rate threshold is exceeded. If this counter continuously increases, first make sure the ICMP packets received are genuine. If they are, increase the current rate threshold.

ICMP packets dropped (ICPktDr)

ICMP packets dropped because the rate threshold has been exceeded.

Bad ICMP checksum (BadCkSum)

ICMP Fragmentation Needed error messages received with an ICMP checksum error.

PMTU non-first IP fragments (PMTUerr)

ICMP Fragmentation Needed error messages received that were generated by an IP fragment other than the first one.

PMTU Invalid body len received (lvBdyLen)

ICMP Fragmentation Needed error messages received that specified an invalid body length.

PMTU no tcp connection (NoTcpCon)

ICMP Need Fragmentation error messages received for TCP packets. The state of the connection for these packets is not maintained on the NetScaler.

PMTU no udp conection (NoUdpCon)

ICMP Need Fragmentation error messages received for UDP packets. The state of the connection for these packets is not maintained on the NetScaler.

PMTU invalid tcp seqno recvd (InvSeqNo)

ICMP Fragmentation Needed error messages received for packets that contain an invalid TCP address.

Invalid next MTU value recvd (lvNxtMTU)

ICMP Fragmentation Needed error messages received in which the Maximum Transmission Unit (MTU) for the next hop is out of range. The range for the MTU is 576-1500.

Next MTU > Current MTU (BigNxMTU)

ICMP Fragmentation Needed error messages received in which the value for the next MTU is higher than that of the current MTU.

PMTU Invalid protocol recvd (IvPrtRx)

ICMP Fragmentation Needed error messages received that contain a protocol other than TCP and UDP.

PMTU IP check sum error (CkSumErr)

ICMP Fragmentation Needed error messages received with an IP checksum error.

PMTU pcb with no link (NoLnkErr)

ICMP Fragmentation Needed error messages received on a Protocol Control Block (PCB) with no link. The PCB maintains the state of the connection.

PMTU Discovery not enabled (PMTUdis)

ICMP Need Fragmentation error messages received when the PMTU Discovery mode is not enabled.

protocol icmpv6

The following operations can be performed on "protocol icmpv6":

stat protocol icmpv6

Displays statistics of the ICMPv6 protocol.

Synopsys

```
stat protocol icmpv6 [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

ICMPv6 packets received (icmpv6RxPkts)

ICMPv6 packets received.

ICMPv6 bytes received (icmpv6RxBytes)

Bytes of ICMPv6 data received.

ICMPv6 packets transmitted (icmpv6TxPkts)

ICMPv6 packets transmitted.

ICMPv6 bytes transmitted (icmpv6TxBytes)

Bytes of ICMPv6 data transmitted.

ICMPv6 NA packets received (icmpv6RxNa)

ICMPv6 neighbor advertisement packets received. These packets are received in response to a neighbor solicitation message sent out by this node, or if the link layer address of a neighbor has changed.

ICMPv6 NS packets received (icmpv6RxNs)

ICMPv6 neighbor solicitation packets received. These packets are received if the link layer address of a neighbor has changed, or in response to a neighbor solicitation message sent out by this node.

ICMPv6 RA packets received (icmpv6RxRa)

ICMPv6 router advertisement packets received. These are received at defined intervals or in response to a router solicitation message.

ICMPv6 RS packets received (icmpv6RxRs)

ICMPv6 router solicitation packets received. These could be sent by a neighboring router to initiate address resolution.

ICMPv6 Echo Request packets received (icmpv6RxEchoReq)

ICMPv6 Ping Echo Request packets received.

ICMPv6 Echo Reply packets received (icmpv6RxEchoReply)

ICMPv6 Ping Echo Reply packets received.

ICMPv6 NA packets transmitted (icmpv6TxNa)

ICMPv6 neighbor advertisement packets transmitted. These packets are sent in response to a neighbor solicitation packet, or if the link layer address of this node has changed.

ICMPv6 NS packets transmitted (icmpv6TxNs)

ICMPv6 neighbor solicitation packets transmitted. These packets are sent to get the link layer addresses of neighboring nodes or to confirm that they are reachable.

ICMPv6 RA packets transmitted (icmpv6TxRa)

ICMPv6 router advertisement packets transmitted. These packets are sent at regular intervals or in response to a router solicitation packet from a neighbor.

ICMPv6 RS packets transmitted (icmpv6TxRs)

ICMPv6 router solicitation packets transmitted. These packets are sent to request neighboring routers to generate router advertisements immediately rather than wait for the next defined time.

ICMPv6 Echo Request packets transmitted (icmpv6TxEchoReq)

ICMPv6 Ping Echo Request packets transmitted.

ICMPv6 Echo Reply packets transmitted (icmpv6TxEchoReply)

ICMP Ping Echo Reply packets transmitted.

ICMPv6 RA error packets (Error in RA packet)

ICMPv6 router advertisement error packets received that contain an error in the header, such as an incorrect source IP address, destination IP address, or packet length.

ICMPv6 NA error packets (Error in NA packet)

ICMPv6 neighbor advertisement error packets received that contain an error in the header, such as an incorrect source IP address, destination IP address, or packet length.

ICMPv6 NS error packets (Error in NS packet)

ICMPv6 neighbor solicitation error packets received that contain an error in the header, such as an incorrect source IP address, destination IP address, or packet length.

ICMPv6 bad checksum (Cksumerr)

Packets received with an ICMPv6 checksum error.

unsupported ICMPv6 packets (icmpv6Unspt)

ICMPv6 packets received that are not supported by the NetScaler.

Rate threshold exceeded packets (icmpv6thslid)

Packets dropped because the default threshold of 100 requests per 10 milliseconds has been exceeded.

This is a configurable value using the set rateControl command.

protocol ip

The following operations can be performed on "protocol ip":

stat protocol ip

Displays statistics of the IP protocol.

Synopsys

```
stat protocol ip [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

IP packets received (IPPktRx)

IP packets received.

IP bytes received (IPbRx)

Bytes of IP data received.

IP packets transmitted (IPPktTx)

IP packets transmitted.

IP bytes transmitted (IPbTx)

Bytes of IP data transmitted.

Megabits received (IPMbRx)

Megabits of IP data received.

Megabits transmitted (IPMbTx)

Megabits of IP data transmitted.

Total routed IP packets (IPRoutedPkts)

Total routed packets.

Total routed IP Mbits (IPRoutedMbits)

Total routed Mbits

IP fragments received (IPFragRx)

IP fragments received.

Successful reassembly (reasSucc)

Fragmented IP packets successfully reassembled on the NetScaler.

Reassembly attempted (reasAtmp)

IP packets that the NetScaler attempts to reassemble. If one of the fragments is missing, the whole packet is dropped.

IP address lookups (IpLkUp)

IP address lookups performed by the NetScaler. When a packet is received on a non-established session, the NetScaler checks if the destination IP address is one of the NetScaler owned IP addresses.

IP address lookup failure (IpLkFail)

IP address lookups performed by the NetScaler that have failed because the destination IP address of the packet does not match any of the NetScaler owned IP addresses.

UDP fragments forwarded (udpFgFwd)

UDP fragments forwarded to the client or the server.

TCP fragments forwarded (tcpFgFwd)

TCP fragments forwarded to the client or the server.

Fragmentation packets created (frgPktCr)

Fragmented packets created by the NetScaler.

Bad IP checksums (badCksum)

Packets received with an IP checksum error.

Unsuccessful reassembly (reasFail)

Packets received that could not be reassembled. This can occur when there is a checksum failure, an identification field mismatch, or when one of the fragments is missing.

Reassembled data too big (reasBig)

Packets received for which the reassembled data exceeds the Ethernet packet data length of 1500 bytes.

Zero fragment length received (zeroLen)

Packets received with a fragment length of 0 bytes.

Duplicate fragments received (dupFrag)

Duplicate IP fragments received. This can occur when the acknowledgement was not received within the expected time.

Out of order fragment received (oooFrag)

Fragments received that are out of order.

Unknown destination received (UnkDst)

Packets received in which the destination IP address was not reachable or not owned by the NetScaler.

Bad Transport (badTran)

Packets received in which the protocol specified in the IP header is unknown to the NetScaler.

VIP down (vipDown)

Packets received for which the VIP is down. This can occur when all the services bound to the VIP are down or the VIP is manually disabled.

Fix header failure (hdrFail)

Packets received that contain an error in one or more components of the IP header.

TTL expired during transit (ttlExp)

Packets for which the time-to-live (TTL) expired during transit. These packets are dropped.

max non-TCP clients (maxClt)

Attempts to open a new connection to a service for which the maximum limit has been exceeded. Default value, 0, applies no limit.

Unknown services (UnkSvc)

Packets received on a port or service that is not configured.

land-attacks (LndAtk)

Land-attack packets received. The source and destination addresses are the same.

Invalid IP header size (errHdrSz)

Packets received in which an invalid data length is specified, or the value in the length field and the actual data length do not match. The range for the Ethernet packet data length is 0-1500 bytes.

Invalid IP packet size (errPktLen)

Total number of packets received by NetScaler with invalid IP packet size.

Truncated IP packet (trIP)

Truncated IP packets received. An overflow in the routers along the path can truncate IP packets.

Truncated non-IP packet (trNonIp)

Truncated non-IP packets received.

ZERO next hop (zrNxtHop)

Packets received that contain a 0 value in the next hop field. These packets are dropped.

Packets with len > 1514 rcvd (BadLenTx)

Packets received with a length greater than the normal maximum transmission unit of 1514 bytes.

Packets with bad MAC sent (BadMacTx)

IP packets transmitted with a bad MAC address.

protocol ipv6

The following operations can be performed on "protocol ipv6":

stat protocol ipv6

Displays statistics of the IPv6 protocol.

Synopsys

```
stat protocol ipv6 [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

IPv6 packets received (ipv6RxPkts)

IPv6 packets received.

IPv6 bytes received (ipv6RxBytes)

Bytes of IPv6 data received.

IPv6 packets transmitted (ipv6TxPkts)

IPv6 packets transmitted

IPv6 bytes transmitted (ipv6TxBytes)

Bytes of IPv6 data transmitted.

Total Routed IPv6 packets (ipv6RoutePkts)

IPv6 packets routed.

Total Routed IPv6 Mbits (ipv6RouteMbits)

IPv6 total Mbits.

IPv6 Fragments received. (ipv6FragRxPkts)

IPv6 fragments received.

TCP Fragments reassembled. (ipv6FragTcpReass)

TCP fragments processed after reassembly.

UDP Fragments reassembled. (ipv6FragUdpReass)

UDP fragments processed after reassembly.

IPv6 Fragments processed without reassembly. (ipv6FragPktsProcessNoReass)

IPv6 fragments processed without reassembly.

IPv6 Fragments bridged. (ipv6FragPktsForward)

IPv6 fragments forwarded to the client or server without reassembly.

IPv6 error hdr packets (RxErrHdr)

Packets received that contain an error in one or more components of the IPv6 header.

IPv6 unsupported next header (Errnxthdr)

Packets received that contain an unsupported next header. The supported next headers are TCP, ICMP, UDP, OSPF, and FRAGMENT.

IPv6 Land-attacks (land attack)

Land-attack packets received. The source and destination addresses are the same. If not dropped, these packets can lock up the appliance.

Reassembled data too big (AssembledPktTooBig)

Packets received for which the reassembled data exceeds the Ethernet packet data length of 1500 bytes.

Zero fragment length received (ZeroLenFramentedPkt)

Packets received with a fragment length of 0 bytes.

ICMPv6 NA packets received

Number of ICMPv6 NA packets received by NetScaler (OBSOLETE).

ICMPv6 NS packets received

Number of ICMPv6 NS packets received by NetScaler (OBSOLETE).

ICMPv6 NA packets transmitted

Number of ICMPv6 NA packets transmitted by NetScaler (OBSOLETE).

ICMPv6 NS packets transmitted

Number of ICMPv6 NS packets transmitted by NetScaler (OBSOLETE).

ICMPv6 RA packets received

Number of ICMPv6 RA packets received by NetScaler (OBSOLETE).

ICMPv6 RS packets transmitted

Number of ICMPv6 RS packets transmitted by NetScaler (OBSOLETE).

ICMPv6 packets received

Number of ICMPv6 packets received by NetScaler (OBSOLETE).

ICMPv6 packets transmitted

Number of ICMPv6 packets transmitted by NetScaler (OBSOLETE).

IPv6 error hdr packets

Number of erroneous header packets received (OBSOLETE).

IPv6 error packets

Number of erroneous packets received (OBSOLETE).

IPv6 bad checksum

Number of bad checksum packets received (OBSOLETE).

ICMPv6 error packets

Number of erroneous ICMPv6 packets received (OBSOLETE).

unsupported ICMPv6 packets

Number of ICMPv6 unsupported packets received (OBSOLETE).

Rate threshold exceeded packets

Number of ICMPv6 packets dropped for rate threshold exceeded (OBSOLETE).

protocol tcp

The following operations can be performed on "protocol tcp":

stat protocol tcp

Displays statistics of the TCP protocol.

Synopsys

```
stat protocol tcp [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

Server active connections (ActSvrCo)

Connections to a server currently responding to requests.

Opening server connections (SvrCxO)

Server connections in the Opening state, which indicates that the handshakes are not yet complete.

Opening client connections (CltCxO)

Client connections in the Opening state, which indicates that the handshakes are not yet complete.

Established client connections (CltCxE)

Current client connections in the Established state, which indicates that data transfer can occur between the NetScaler and the client.

Established server connections (SvrCxE)

Current server connections in the Established state, which indicates that data transfer can occur between the NetScaler and the server.

TCP packets received (TCPPktRx)

TCP packets received.

TCP bytes received (TCPbRx)

Bytes of TCP data received.

TCP packets transmitted (TCPPktTx)

TCP packets transmitted.

TCP bytes transmitted (TCPbTx)

Bytes of TCP data transmitted.

All client connections (CltCx)

Client connections, including connections in the Opening, Established, and Closing state.

Closing client connections (CltCxCI)

Client connections in the Closing state, which indicates that the connection termination process has initiated but is not complete.

Opened client connections (TotCltO)

Client connections opened by the NetScaler since startup (after three-way handshake). This counter is reset when the NetScaler is restarted.

All server connections (SvrCx)

Server connections, including connections in the Opening, Established, and Closing state.

Closing server connections (SvrCxCI)

Server connections in the Closing state, which indicates that the connection termination process has initiated but is not complete.

Opened server connections (TotSvrO)

Server connections initiated by the NetScaler since startup. This counter is reset when the NetScaler is restarted.

Surge queue (SQlen)

Connections in the surge queue. When the NetScaler cannot open a connection to the server, for example when maximum connections have been reached, the NetScaler queues these requests.

Spare connections (SpConn)

Spare connections available. To save time and resources in establishing another connection for a new client, the connection on the server is not closed after completing the request from the first client and is available for serving future requests.

Client idle flushed (ZomCltF)

Client connections that are flushed because the client has been idle for some time.

Client half opened flushed (ZCltFHo)

Half-opened client connections that are flushed because the three-way handshakes are not complete.

Client active half closed flushed (ZCltFAhc)

Active half-closed client connections that are flushed because the client has closed the connection and there has been no activity on the connection.

Client passive half closed flushed (ZCltFPhc)

Passive half-closed client connections that are flushed because the NetScaler has closed the connection and there has been no activity on the connection.

Server idle connections flushed (ZSvrF)

Server connections that are flushed because there have been no client requests in the queue for some time.

Server half opened flushed (ZSvrFHo)

Half-opened server connections that are flushed because the three-way handshakes are not complete.

Server active half closed flushed (ZSvrFAhc)

Active half-closed server connections that are flushed because the server has closed the connection and there has been no activity on the connection.

Server passive half closed flushed (ZSrvFPhc)

Passive half-closed server connections that are flushed because the NetScaler has closed the connection and there has been no activity on the connection.

Zombie cleanup calls (ZmbCall)

Times the Zombie cleanup function is called. Every time a connection is flushed, it is marked for cleanup. The Zombie cleanup function clears all these connections at predefined intervals.

SYN packets received (TCPSYN)

SYN packets received

Server probes (SYNProbe)

Probes from the NetScaler to a server. The NetScaler sends a SYN packet to the server to check its availability and expects a SYN_ACK packet from the server before a specified response timeout.

FIN packets from server (SvrFin)

FIN packets received from the server.

FIN packets from client (CltFin)

FIN packets received from the clients.

Time wait to SYN (WaToSyn)

SYN packets (packets used to initiate a TCP connection) received on connections that are in the TIME_WAIT state. Packets cannot be transferred on a connection in this state.

Data in TIME_WAIT (WaDat)

Bytes of data received on connections that are in the TIME_WAIT state. Data cannot be transferred on a connection that is in this state.

SYN packets held (SYNHeld)

SYN packets held on the NetScaler that are waiting for a server connection.

SYN packets flushed (SYNFlush)

SYN packets flushed on the NetScaler because of no response from the server for three or more seconds.

TIME_WAIT connections closed (FinWaitC)

Connections closed on the NetScaler because the number of connections in the TIME_WAIT state has exceeded the default value of 7000.

Bad TCP checksum (TCPBadCk)

Packets received with a TCP checksum error.

Data after FIN (TCPDfFin)

Packets received following a connection termination request. This error is usually caused by a reordering of packets during transmission.

SYN in SYN_RCVD state (TCPSYNRv)

SYN packets received on a connection that is in the SYN_RCVD state. A connection goes into the SYN_RCVD state after receiving a SYN packet.

SYN in ESTABLISHED state (TCPSYNEs)

SYN packets received on a connection that is in the ESTABLISHED state. A SYN packet is not expected on an ESTABLISHED connection.

SYN_SENT incorrect ACK packet (TCPBadAk)

Incorrect ACK packets received on a connection that is in the SYN_SENT state. An incorrect ACK packet is the third packet in the three-way handshake that has an incorrect sequence number.

RST packets received (TCPRST)

Reset packets received from a client or a server.

RST on not ESTABLISHED (TCPRSTNE)

Reset packets received on a connection that is not in the ESTABLISHED state.

RST out of window (TCPRSTOW)

Reset packets received on a connection that is out of the current TCP window.

RST in TIME_WAIT (TCPRSTTi)

Reset packets received on a connection that is in the TIME_WAIT state. Packets cannot be transferred on a connection in the TIME_WAIT state.

Server out of order packets (SvrOOO)

Out of order TCP packets received from a server.

Client out of order packets (CltOOO)

Out of order TCP packets received from a client.

TCP hole on client connection (Clthole)

TCP holes created on a client connection. When out of order packets are received from a client, a hole is created on the NetScaler for each group of missing packets.

TCP hole on server connection (SvrHole)

TCP holes created on a server connection. When out of order packets are received from a server, a hole is created on the NetScaler for each group of missing packets.

Seq number SYN cookie reject (CSeqRej)

SYN cookie packets rejected because they contain an incorrect sequence number.

Signature SYN cookie reject (CSigRej)

SYN cookie packets rejected because they contain an incorrect signature.

Seq number SYN cookie drop (CSigDrp)

SYN cookie packets dropped because the sequence number specified in the packets is outside the current window.

MSS SYN cookie reject (CMssRej)

SYN cookie packets rejected because the maximum segment size (MSS) specified in the packets is incorrect.

Any IP port allocation failure (PortFal)

Port allocations that have failed on a mapped IP address because the maximum limit of 65536 has been exceeded.

IP port allocation failure (PortFall)

Port allocations that have failed on a subnet IP address or vserver IP address because the maximum limit of 65536 has been exceeded.

Stray packets (StrayPkt)

Number of stray or misrouted packets.

RST packets sent (SentRst)

Reset packets sent to a client or a server.

Bad state connections (BadConn)

Connections that are not in a valid TCP state.

RST threshold dropped (RstThre)

Reset packets dropped because the default threshold of 100 resets per 10 milliseconds has been exceeded. This is a configurable value using the set rateControl command.

Packets out of window (OOWPkt)

Packets received that are out of the current advertised window.

SYNs dropped (Congestion) (SynCng)

SYN packets dropped because of network congestion.

Client retransmissions (TCPClRe)

Packets retransmitted by a client. This usually occurs because the acknowledgement from the NetScaler has not reached the client.

Full packet retransmissions (TCPFullRe)

Full packets retransmitted by the client or the server.

SYN packet retries (TCPSYNRe)

SYN packets resent to a server.

SYN packets timeout (TCPSYNG)

Attempts to establish a connection on the NetScaler that timed out.

TCP retransmission (Retr)

TCP packets retransmitted. The NetScaler attempts to retransmit the packet up to seven times, after which it resets the other half of the TCP connection.

1st retransmission (1stRetr)

Packets retransmitted once by the NetScaler.

3rd retransmission (3rdRetr)

Packets retransmitted three times by the NetScaler.

5th retransmission (5thRetr)

Packets retransmitted five times by the NetScaler.

7th retransmission (7thRetr)

Packets retransmitted seven times by the NetScaler. If this fails, the NetScaler terminates the connection.

Fast retransmits (FastRetr)

TCP packets on which the NetScaler performs a fast retransmission in response to three duplicate acknowledgements or a partial acknowledgement. The NetScaler assumes that the packet is lost and retransmits the packet before its time-out.

Server retransmissions (TCPSvrRe)

Packets retransmitted by a server. This usually occurs because the acknowledgement from the NetScaler has not reached the server.

Partial packet retransmissions (TCPParRe)

Partial packet retransmits by a client or server due to congestion on the connection. This usually occurs because the window advertised by the NetScaler is not big enough to hold the full packet.

FIN packet retries (TCPFINRe)

FIN packets resent to a server or a client.

FIN packets timeout (TCPFING)

Connections that were timed out by the NetScaler because of not receiving the ACK packet after retransmitting the FIN packet four times.

2nd retransmission (2ndRetr)

Packets retransmitted twice by the NetScaler.

4th retransmission (4thRetr)

Packets retransmitted four times by the NetScaler.

6th retransmission (6thRetr)

Packets retransmitted six times by the NetScaler.

TCP retransmission giveup (RetrG)

Number of times NetScaler terminates a connection after retransmitting the packet seven times on that connection. Retransmission happens when receiving end doesn't acknowledge the packet.

TCP level cip failure (CltHdrEr)

Number of times TCP level client header insertion failure

protocol udp

The following operations can be performed on "protocol udp":

stat protocol udp

Displays statistics of the UDP protocol.

Synopsys

```
stat protocol udp [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

Packets received (UDPPktRx)

Total number of UDP packets received.

Bytes received (UDPbRx)

Total number of UDP data received in bytes.

Packets transmitted (UDPPktTx)

Total number of UDP packets transmitted.

Bytes transmitted (UDPbTx)

Total number of UDP data transmitted in bytes.

Current rate threshold (UDPThs)

Limit for UDP packets handled every 10 milliseconds. Default value, 0, applies no limit.

This is a configurable value using the set rateControl command.

Unknown service (UDPUnSvc)

Stray UDP packets dropped due to no configured listening service.

Bad UDP checksum (UDPBadCkSum)

Packets received with a UDP checksum error.

Rate threshold exceeded (UDPRtEx)

Number of times the UDP rate threshold is exceeded. If this counter continuously increases, first make sure the UDP packets received are genuine.

If they are, increase the current rate threshold. This is a configurable value using the `set rateControl` command.

QOS Commands

The entities on which you can perform NetScaler CLI operations:

- qos
- qos stats

qos

The following operations can be performed on "qos":

stat qos

Display QoS statistics.

Synopsys

```
stat qos [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

IPC messages sent from QoS (ipcsent)

IPC messages sent from qos system.

IPC messages QoS failed to send (ipcfail)

IPC messages failed to send from qos system.

IPC messages QoS received (ipcrcvd)

IPC messages received by qos.

IPC messages sent to QoS (pe2qsent)

IPC messages sent to qos system.

IPC messages failed to send QoS (pe2qfail)

IPC messages failed to send to qos system.

IPC messages received from QoS (pe2qrecv)

IPC messages received from qos system.

Bytes QoS marked for drop (bytsdrop)

Bytes QoS marked for drop

QoS bytes sent not classified (bytsntnc)

Bytes scheduled by QoS that were not classified.

QoS bytes dropped no connection (bytdrpnc)

Bytes dropped by QoS when no connection was found.

Packets sent to QoS (qosinpkt)

Packets sent to QoS for scheduling.

Packets from QoS to be sent (qosotpkt)

Packets from QoS to be sent

Packets Dropped by QoS (qosdrpkt)

Packets Dropped by QoS.

Classified source MAC rewritten (qosrwmac)

Number of packets with inband classification in source MAC.

QoS packets unclassified (qosucclas)

Number of packets without classification.

QoS packets classified (qosclas)

Number of packets with classification.

QoS learned true MAC (qoslm)

QoS learned true MAC

QoS Input Bytes (qosib)

Bytes sent to QoS for scheduling

QoS Output Bytes (qosob)

Bytes received from QoS to be sent

QoS Free Held List (qosfc)

No. more packets QoS can hold onto.

QoS Link 00 Bytes Sent (qosl00sd)

QoS bytes sent on Link 00

QoS Link 00 Bytes Dropped (qosl00dr)

QoS bytes dropped on Link 00

QoS Link 01 Bytes Sent (qosl01sd)

QoS bytes sent on Link 01

QoS Link 01 Bytes Dropped (qosl01dr)

QoS bytes dropped on Link 01

QoS Link 02 Bytes Sent (qosl02sd)

QoS bytes sent on Link 02

QoS Link 02 Bytes Dropped (qosl02dr)

QoS bytes dropped on Link 02

QoS Link 03 Bytes Sent (qosl03sd)

QoS bytes sent on Link 03

QoS Link 03 Bytes Dropped (qosl03dr)

QoS bytes dropped on Link 03

QoS Link 04 Bytes Sent (qosl04sd)

QoS bytes sent on Link 04

QoS Link 04 Bytes Dropped (qosl04dr)

QoS bytes dropped on Link 04

QoS Link 05 Bytes Sent (qosl05sd)

QoS bytes sent on Link 05

QoS Link 05 Bytes Dropped (qosl05dr)

QoS bytes dropped on Link 05

QoS Link 06 Bytes Sent (qosl06sd)

QoS bytes sent on Link 06

QoS Link 06 Bytes Dropped (qosl06dr)

QoS bytes dropped on Link 06

QoS Link 07 Bytes Sent (qosl07sd)

QoS bytes sent on Link 07

QoS Link 07 Bytes Dropped (qosl07dr)

QoS bytes dropped on Link 07

QoS Link 08 Bytes Sent (qosl08sd)

QoS bytes sent on Link 08

QoS Link 08 Bytes Dropped (qosl08dr)

QoS bytes dropped on Link 08

QoS Link 09 Bytes Sent (qosl09sd)

QoS bytes sent on Link 09

QoS Link 09 Bytes Dropped (qosl09dr)

QoS bytes dropped on Link 09

QoS Link 10 Bytes Sent (qosl10sd)

QoS bytes sent on Link 10

QoS Link 10 Bytes Dropped (qosl10dr)

QoS bytes dropped on Link 10

qos stats

The following operations can be performed on "qos stats":

show qos stats

show qos stats is an alias for stat qos

Synopsys

show qos stats - alias for 'stat qos'

Responder Commands

The entities on which you can perform NetScaler CLI operations:

- responder action
- responder global
- responder htmlpage
- responder param
- responder policy
- responder policylabel

responder action

The following operations can be performed on "responder action":

add | **rm** | **set** | **unset** | **show** | **rename**

add responder action

Creates a responder action, which specifies how to respond to a request.

Synopsys

```
add responder action <name> <type> (<target> | <htmlpage>) [-bypassSafetyCheck ( YES | NO )] [-comment <string>]
```

Arguments

name

Name for the responder action. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) hash (#), space (), at (@), equals (=), colon (:), and underscore characters. Can be changed after the responder policy is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my responder action" or 'my responder action').

type

Type of responder action. Available settings function as follows:

- * **respondwith** <target> - Respond to the request with the expression specified as the target.
- * **respondwithhtmlpage** - Respond to the request with the uploaded HTML page object specified as the target.
- * **redirect** - Redirect the request to the URL specified as the target.
- * **sqlresponse_ok** - Send an SQL OK response.
- * **sqlresponse_error** - Send an SQL ERROR response.

Possible values: noop, respondwith, redirect, respondwithhtmlpage, sqlresponse_ok, sqlresponse_error

target

Expression specifying what to respond with. Typically a URL for redirect policies or a default-syntax expression. In addition to NetScaler default-syntax expressions that refer to information in the request, a stringbuilder expression can contain text and HTML, and simple escape codes that define new lines and paragraphs. Enclose each stringbuilder expression element (either a NetScaler default-syntax expression or a string) in double quotation marks. Use the plus (+) character to join the elements.

Examples:

1) Respondwith expression that sends an HTTP 1.1 200 OK response:

```
"HTTP/1.1 200 OK\r\n\r\n"
```

2) Redirect expression that redirects user to the specified web host and appends the request URL to the redirect.

```
"http://backupsite2.com" + HTTP.REQ.URL
```

3) Respondwith expression that sends an HTTP 1.1 404 Not Found response with the request URL included in the response:

```
"HTTP/1.1 404 Not Found\r\n\r\n" + "HTTP.REQ.URL.HTTP_URL_SAFE" + "does not exist on the web server."
```

The following requirement applies only to the NetScaler CLI:

Enclose the entire expression in single quotation marks. (NetScaler default expression elements should be included inside the single quotation marks for the entire expression, but do not need to be enclosed in double quotation marks.)

htmlpage

For `respondwithhtmlpage` policies, name of the HTML page object to use as the response. You must first import the page object.

bypassSafetyCheck

Bypass the safety check, allowing potentially unsafe expressions. An unsafe expression in a response is one that contains references to request elements that might not be present in all requests. If a response refers to a missing request element, an empty string is used instead.

Possible values: YES, NO

Default value: NO

comment

Comment. Any type of information about this responder action.

Example

```
1) add responder action act1 respondwith "\\\\"HTTP/1.1 200 OK\\\\\\r\\\\\\n\\\\\\r\\\\\\n\\\\\\n\\\\\\n\\\\\\n\\\\\\n" :
```

rm responder action

Removes the specified responder action.

Synopsis

```
rm responder action <name>
```

Arguments

name

Name of the responder action to remove.

Example

```
rm responder action act_before
```

set responder action

Modifies the specified parameters of a responder action.

Synopsis

```
set responder action <name> [-target <string> [-bypassSafetyCheck ( YES | NO )]] [-htmlpage <string>] [-comment <string>]
```

Arguments

name

Name of the responder action to be modified.

target

Expression specifying what to respond with. Typically a URL for redirect policies or a default-syntax expression. In addition to NetScaler default-syntax expressions that refer to information in the request, a stringbuilder expression can contain text and HTML, and simple escape codes that define new lines and paragraphs. Enclose each stringbuilder expression element (either a NetScaler default-syntax expression or a string) in double quotation marks. Use the plus (+) character to join the elements.

Examples:

1) Respondwith expression that sends an HTTP 1.1 200 OK response:

```
"HTTP/1.1 200 OK\r\n\r\n"
```

2) Redirect expression that redirects user to the specified web host and appends the request URL to the redirect.

```
"http://backupsite2.com" + HTTP.REQ.URL
```

3) Respondwith expression that sends an HTTP 1.1 404 Not Found response with the request URL included in the response:

```
"HTTP/1.1 404 Not Found\r\n\r\n" + "HTTP.REQ.URL.HTTP_URL_SAFE" + "does not exist on the web server."
```

The following requirement applies only to the NetScaler CLI:

Enclose the entire expression in single quotation marks. (NetScaler default expression elements should be included inside the single quotation marks for the entire expression, but do not need to be enclosed in double quotation marks.)

bypassSafetyCheck

Bypass the safety check, allowing potentially unsafe expressions. An unsafe expression in a response is one that contains references to request elements that might not be present in all requests. If a response refers to a missing request element, an empty string is used instead.

Possible values: YES, NO

Default value: NO

htmlpage

For respondwithhtmlpage policies, name of the HTML page object to use as the response. You must first import the page object.

comment

Comment. Any type of information about this responder action.

Example

```
1. set responder action act_responder -target 'HTTP.REQ.HEADER(MYURL)' -bypassSafetyCheck
```

unset responder action

Use this command to remove responder action settings. Refer to the set responder action command for meanings of the arguments.

Synopsys

```
unset responder action <name> -comment
```

show responder action

Displays the current settings for the specified responder action. If no action name is provided, displays a list of all responder actions currently configured on the NetScaler appliance, with abbreviated settings.

Synopsys

show responder action [<name>]

Arguments

name

Name of the responder action.

Outputs

stateflag

type

Type of responder action. It can be: (respondwith).

target

Expression specifying what to respond with

htmlpage

Option specifying to respondwith htmlpage

bypassSafetyCheck

The safety check to allow unsafe expressions.

hits

The number of times the action has been taken.

referenceCount

The number of references to the action.

undefHits

The number of times the action resulted in UNDEF.

comment

Comment. Any type of information about this responder action.

builtin

Flag to determine whether responder action is built-in or not

devno

count

Example

```
1. show responder action 2. show responder action act_insert
```

rename responder action

Renames a responder action.

Synopsys

rename responder action <name>@ <newName>@

Arguments

name

Existing name of the responder action.

newName

New name for the responder action.

Must begin with a letter, number, or the underscore character (`_`), and must contain only letters, numbers, and the hyphen (`-`), period (`.`) hash (`#`), space (), at (`@`), equals (`=`), colon (`:`), and underscore characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my responder action" or my responder action').

Example

```
rename responder action oldname newname
```

responder global

The following operations can be performed on "responder global":

[bind](#) | [unbind](#) | [show](#)

bind responder global

Activates the specified responder policy for all requests sent to the NetScaler appliance.

Synopsys

```
bind responder global <policyName> <priority> [<gotoPriorityExpression>] [-type <type>] [-invoke (<labelType>
<labelName>)]
```

Arguments

policyName

Name of the responder policy to activate. If you want to create the policy as well as activate it, specify a name for the responder policy. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) hash (#), space (), at (@), equals (=), colon (:), and underscore characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my responder policy" or 'my responder policy').

priority

Positive integer specifying the priority of the policy. A lower number specifies a higher priority. Must be unique within the label. Policies are evaluated in the order of their priority numbers, and the first policy that matches the request is applied.

Minimum value: 1

Maximum value: 2147483647

gotoPriorityExpression

Expression or other value specifying the next policy to evaluate if the current policy evaluates to TRUE. Specify one of the following values:

- * NEXT - Evaluate the policy with the next higher priority number.
- * END - End policy evaluation.
- * USE_INVOCATION_RESULT - Applicable if this policy invokes another policy label. If the final goto in the invoked policy label has a value of END, the evaluation stops. If the final goto is anything other than END, the current policy label performs a NEXT.
- * A default syntax or classic expression that evaluates to a number.

If you specify an expression, the number to which it evaluates determines the next policy to evaluate, as follows:

- * If the expression evaluates to a higher numbered priority, the policy with that priority is evaluated next.
- * If the expression evaluates to the priority of the current policy, the policy with the next higher numbered priority is evaluated next.
- * If the expression evaluates to a number that is larger than the largest numbered priority, policy evaluation ends.

An UNDEF event is triggered if:

- * The expression is invalid.

- * The expression evaluates to a priority number that is smaller than the current policy's priority number.
- * The expression evaluates to a priority number that is between the current policy's priority number (say, 30) and the highest priority number (say, 100), but does not match any configured priority number (for example, the expression evaluates to the number 85). This example assumes that the priority number increments by 10 for every successive policy, and therefore a priority number of 85 does not exist in the policy label.

type

Bind point to which to bind the policy. Available settings function as follows:

- * REQ_OVERRIDE - Request override. Binds the policy to the priority request queue.
- * REQ_DEFAULT - Binds the policy to the default request queue.
- * OTHERTCP_REQ_OVERRIDE - Binds the policy to the non-HTTP TCP priority request queue.
- * OTHERTCP_REQ_DEFAULT - Binds the policy to the non-HTTP TCP default request queue.
- * SIPUDP_REQ_OVERRIDE - Binds the policy to the SIP UDP priority response queue.
- * SIPUDP_REQ_DEFAULT - Binds the policy to the SIP UDP default response queue.
- * MSSQL_REQ_OVERRIDE - Binds the policy to the Microsoft SQL priority response queue.
- * MSSQL_REQ_DEFAULT - Binds the policy to the Microsoft SQL default response queue.
- * MYSQL_REQ_OVERRIDE - Binds the policy to the MySQL priority response queue.
- * MYSQL_REQ_DEFAULT - Binds the policy to the MySQL default response queue.
- * NAT_REQ_OVERRIDE - Binds the policy to the NAT priority request queue.
- * NAT_REQ_DEFAULT - Binds the policy to the NAT default request queue.

Possible values: REQ_OVERRIDE, REQ_DEFAULT, OVERRIDE, DEFAULT, OTHERTCP_REQ_OVERRIDE, OTHERTCP_REQ_DEFAULT, SIPUDP_REQ_OVERRIDE, SIPUDP_REQ_DEFAULT, MSSQL_REQ_OVERRIDE, MSSQL_REQ_DEFAULT, MYSQL_REQ_OVERRIDE, MYSQL_REQ_DEFAULT, NAT_REQ_OVERRIDE, NAT_REQ_DEFAULT, DIAMETER_REQ_OVERRIDE, DIAMETER_REQ_DEFAULT

invoke

If the current policy evaluates to TRUE, terminate evaluation of policies bound to the current policy label, and then forward the request to the specified virtual server or evaluate the specified policy label.

labelType

Type of invocation, Available settings function as follows:

- * vserver - Forward the request to the specified virtual server.
- * policylabel - Invoke the specified policy label.

Possible values: vserver, policylabel

labelName

Name of the policy label to invoke. If the current policy evaluates to TRUE, the invoke parameter is set, and Label Type is policylabel.

Example

```
i) bind responder global pol9 9
```

unbind responder global

Unbind the specified responder policy from responder global.

Synopsys

unbind responder global <policyName> [-type <type>] [-priority <positive_integer>]

Arguments

policyName

Name of the policy to unbind.

type

The bindpoint from which the policy is to be unbound.

Possible values: REQ_OVERRIDE, REQ_DEFAULT, OVERRIDE, DEFAULT, OTHERTCP_REQ_OVERRIDE, OTHERTCP_REQ_DEFAULT, SIPUDP_REQ_OVERRIDE, SIPUDP_REQ_DEFAULT, MSSQL_REQ_OVERRIDE, MSSQL_REQ_DEFAULT, MYSQL_REQ_OVERRIDE, MYSQL_REQ_DEFAULT, NAT_REQ_OVERRIDE, NAT_REQ_DEFAULT, DIAMETER_REQ_OVERRIDE, DIAMETER_REQ_DEFAULT

priority

Priority of the NOPOLICY to be unbound.

Minimum value: 1

Maximum value: 2147483647

Example

```
unbind responder global pol9
```

show responder global

Displays the list of policies bound to the specified responder global bind point. If no bind point is specified, displays a list of all policies bound to responder global.

Synopsys

show responder global [-type <type>]

Arguments

type

Specifies the bind point whose policies you want to display. Available settings function as follows:

- * REQ_OVERRIDE - Request override. Binds the policy to the priority request queue.
- * REQ_DEFAULT - Binds the policy to the default request queue.
- * OTHERTCP_REQ_OVERRIDE - Binds the policy to the non-HTTP TCP priority request queue.
- * OTHERTCP_REQ_DEFAULT - Binds the policy to the non-HTTP TCP default request queue..
- * SIPUDP_REQ_OVERRIDE - Binds the policy to the SIP UDP priority response queue..
- * SIPUDP_REQ_DEFAULT - Binds the policy to the SIP UDP default response queue.
- * MSSQL_REQ_OVERRIDE - Binds the policy to the Microsoft SQL priority response queue..
- * MSSQL_REQ_DEFAULT - Binds the policy to the Microsoft SQL default response queue.
- * MYSQL_REQ_OVERRIDE - Binds the policy to the MySQL priority response queue.
- * MYSQL_REQ_DEFAULT - Binds the policy to the MySQL default response queue.

Possible values: REQ_OVERRIDE, REQ_DEFAULT, OVERRIDE, DEFAULT, OTHERTCP_REQ_OVERRIDE, OTHERTCP_REQ_DEFAULT, SIPUDP_REQ_OVERRIDE, SIPUDP_REQ_DEFAULT, MSSQL_REQ_OVERRIDE, MSSQL_REQ_DEFAULT, MYSQL_REQ_OVERRIDE, MYSQL_REQ_DEFAULT, NAT_REQ_OVERRIDE, NAT_REQ_DEFAULT, DIAMETER_REQ_OVERRIDE, DIAMETER_REQ_DEFAULT

Outputs

stateflag

policyName

Name of the responder policy.

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

invoke

If the current policy evaluates to TRUE, terminate evaluation of policies bound to the current policy label, and then forward the request to the specified virtual server or evaluate the specified policy label.

labelType

Type of invocation, Available settings function as follows:

* vserver - Forward the request to the specified virtual server.

* policylabel - Invoke the specified policy label.

labelName

Name of the policy label to invoke. If the current policy evaluates to TRUE, the invoke parameter is set, and Label Type is policylabel.

flowType

flowtype of the bound responder policy.

numpol

number of polices bound to label.

flags

devno

count

Example

```
show responder global
```

responder htmlpage

The following operations can be performed on "responder htmlpage":

[import](#) | [rm](#) | [update](#) | [show](#)

import responder htmlpage

Imports the specified HTML page to the NetScaler appliance, assigns it the specified name, and stores it in the list of Responder HTML page objects.

Synopsys

```
import responder htmlpage [<src>] <name> [-comment <string>] [-overwrite]
```

Arguments

src

Local path to and name of, or URL \\(protocol, host, path, and file name\\) for, the file in which to store the imported HTML page.

NOTE: The import fails if the object to be imported is on an HTTPS server that requires client certificate authentication for access.

name

Name to assign to the HTML page object on the NetScaler appliance.

comment

Any comments to preserve information about the HTML page object.

overwrite

Overwrites the existing file

Example

```
import responder htmlpage http://www.example.com/page.html my-responder-page
```

rm responder htmlpage

Removes the specified HTML page object.

Synopsys

```
rm responder htmlpage <name>
```

Arguments

name

Name of the HTML page object to remove.

Example

```
rm responder htmlpage <name>
```

update responder htmlpage

Updates the specified HTML page object from the source.

Synopsys

```
update responder htmlpage <name>
```

Arguments

name

Name to assign to the HTML page object on the NetScaler appliance.

Example

```
update responder htmlpage my-responder-page
```

show responder htmlpage

Displays the specified HTML page object. If no HTML page object is specified, lists all HTML page objects on the NetScaler appliance.

Synopsys

```
show responder htmlpage [<name>]
```

Arguments

name

Name of the HTML page object to display.

Outputs

response

Example

```
show responder htmlpage
```

responder param

The following operations can be performed on "responder param":

[set](#) | [unset](#) | [show](#)

set responder param

Sets the default responder undefined action. If an UNDEF event is triggered during policy evaluation and if no undefAction is specified for the current policy, this value is used.

Synopsys

```
set responder param -undefAction <string>
```

Arguments

undefAction

Action to perform when policy evaluation creates an UNDEF condition. Available settings function as follows:

- * NOOP - Send the request to the protected server.
- * RESET - Reset the request and notify the user's browser, so that the user can resend the request.
- * DROP - Drop the request without sending a response to the user.

Default value: "NOOP"

Example

```
set responder param -undefAction RESET
```

unset responder param

Resets the global undefAction to NOOP..Refer to the set responder param command for meanings of the arguments.

Synopsys

```
unset responder param -undefAction
```

Example

```
unset responder param -undefAction
```

show responder param

Displays the default responder undefAction.

Synopsys

```
show responder param
```

Outputs

undefAction

Name of the responder action.

Example

```
show responder param
```


responder policy

The following operations can be performed on "responder policy":

add | **rm** | **set** | **unset** | **show** | **rename** | **stat**

add responder policy

Creates a responder policy, which specifies requests that the NetScaler appliance intercepts and responds to directly instead of forwarding them to a protected server.

Synopsys

```
add responder policy <name> <rule> <action> [<undefAction>] [-comment <string>] [-logAction <string>] [-appflowAction <string>]
```

Arguments

name

Name for the responder policy.

Must begin with a letter, number, or the underscore character (`_`), and must contain only letters, numbers, and the hyphen (`-`), period (`.`) pound (`#`), space (), at (`@`), equals (`=`), colon (`:`), and underscore characters. Can be changed after the responder policy is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my responder policy" or 'my responder policy').

rule

Default syntax expression that the policy uses to determine whether to respond to the specified request.

action

Name of the responder action to perform if the request matches this responder policy. There are also some built-in actions which can be used. These are:

- * NOOP - Send the request to the protected server instead of responding to it.
- * RESET - Reset the client connection by closing it. The client program, such as a browser, will handle this and may inform the user. The client may then resend the request if desired.
- * DROP - Drop the request without sending a response to the user.

undefAction

Action to perform if the result of policy evaluation is undefined (UNDEF). An UNDEF event indicates an internal error condition. Only the above built-in actions can be used.

comment

Any type of information about this responder policy.

logAction

Name of the message log action to use for requests that match this policy.

appflowAction

AppFlow action to invoke for requests that match this policy.

Example

```
i) add responder policy pol9 "HTTP.REQ.HEADER(\\\\"header\\\\" ).CONTAINS(\\\\"qh3\\\\" )" :
```

rm responder policy

Removes the specified responder policy.

Synopsys

```
rm responder policy <name>
```

Arguments

name

Name of the responder policy to remove.

Example

```
rm responder policy pol9
```

set responder policy

Modifies the rule or action portion of the specified responder policy.

Synopsys

```
set responder policy <name> [-rule <expression>] [-action <string>] [-undefAction <string>] [-comment <string>] [-logAction <string>] [-appflowAction <string>]
```

Arguments

name

Name of the responder policy.

rule

Default syntax expression that the policy uses to determine whether to respond to the specified request.

action

Name of the responder action to perform if the request matches this responder policy. There are also some built-in actions which can be used. These are:

- * NOOP - Send the request to the protected server instead of responding to it.

- * RESET - Reset the client connection by closing it. The client program, such as a browser, will handle this and may inform the user. The client may then resend the request if desired.

- * DROP - Drop the request without sending a response to the user.

undefAction

Action to perform if the result of policy evaluation is undefined (UNDEF). An UNDEF event indicates an internal error condition. Only the above built-in actions can be used.

comment

Any type of information about this responder policy.

logAction

Name of the message log action to use for requests that match this policy.

appflowAction

AppFlow action to invoke for requests that match this policy.

Example

```
set responder policy pol9 -rule "HTTP.REQ.HEADER(\\\\\\"header\\\\\\").CONTAINS(\\\\\\"qh2\\\\\\")"
```

unset responder policy

Removes the settings of an existing responder policy. Attributes for which a default value is available revert to their default values. See the set responder policy command for descriptions of the parameters..Refer to the set responder policy command for meanings of the arguments.

Synopsys

```
unset responder policy <name> [-undefAction] [-comment] [-logAction] [-appflowAction]
```

Example

```
unset responder policy respol9 -undefAction
```

show responder policy

Displays the current settings for the specified responder policy. If no policy name is specified, displays a list of all responder policies currently configured on the NetScaler appliance, with abbreviated settings.

Synopsys

```
show responder policy [<name>] show responder policy stats - alias for 'stat responder policy'
```

Arguments

name

Name of the responder policy for which to display settings.

Outputs

stateflag

rule

Rule of the policy.

action

Responder action associated with the policy.

undefAction

UNDEF action associated with the policy.

hits

Number of hits.

undefHits

Number of policy UNDEF hits.

activePolicy

Indicates whether policy is bound or not.

boundTo

Location where policy is bound

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

labelType

Type of policy label invocation.

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

comment

Any type of information about this responder policy.

logAction

Name of the message log action to use for requests that match this policy.

bindPolicyType**vserverType****appflowAction**

AppFlow action to invoke for requests that match this policy.

builtin

Flag to determine if responder policy is built-in or not

devno**count**

Example

```
show responder policy
```

rename responder policy

Renames the specified responder policy.

Synopsis

```
rename responder policy <name>@ <newName>@
```

Arguments

name

Existing name of the responder policy.

newName

New name for the responder policy. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) hash (#), space (), at (@), equals (=), colon (:), and underscore characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my responder policy" or 'my responder policy').

Example

```
rename responder policy oldname newname
```

stat responder policy

Displays statistics for all responder policies currently configured on the NetScaler appliance, or detailed statistics for the specified policy.

Synopsys

```
stat responder policy [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

name

Name of the responder policy for which to show detailed statistics.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Policy hits (Hits)

Number of hits on the policy

Policy undef hits (Undefhits)

Number of undef hits on the policy

responder policylabel

The following operations can be performed on "responder policylabel":

add | **rm** | **bind** | **unbind** | **show** | **stat** | **rename**

add responder policylabel

Creates a user-defined responder policy label, to which you can bind policies. A policy label is a tool for evaluating a set of policies in a specified order. By using a policy label, you can configure the responder feature to choose the next policy, invoke a different policy label, or terminate policy evaluation completely by looking at whether the previous policy evaluated to TRUE or FALSE.

Synopsys

```
add responder policylabel <labelName> [-policylabeltype <policylabeltype>] [-comment <string>]
```

Arguments

labelName

Name for the responder policy label. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) hash (#), space (), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after the responder policy label is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my responder policy label" or my responder policy label').

policylabeltype

Type of responses sent by the policies bound to this policy label. Types are:

- * HTTP - HTTP responses.
- * OTHERTCP - NON-HTTP TCP responses.
- * SIP_UDP - SIP responses.
- * MYSQL - SQL responses in MySQL format.
- * MSSQL - SQL responses in Microsoft SQL format.
- * NAT - NAT response.

Possible values: HTTP, OTHERTCP, SIP_UDP, MYSQL, MSSQL, NAT, DIAMETER

Default value: HTTP

comment

Any comments to preserve information about this responder policy label.

Example

```
add responder policylabel resp_lab
```

rm responder policylabel

Removes a responder policy label.

Synopsys

```
rm responder policylabel <labelName>
```

Arguments

labelName

Name of the responder policy label to remove.

Example

```
rm responder policylabel resp_lab
```

bind responder policylabel

Binds the specified responder policy to the specified policy label.

Synopsys

```
bind responder policylabel <labelName> <policyName> <priority> [<gotoPriorityExpression>] [-invoke (<labelType>  
<labelName>)]
```

Arguments

labelName

- * If labelType is policylabel, name of the policy label to invoke.
- * If labelType is reqvserver or resvserver, name of the virtual server.

policyName

Name of the policy to bind to the responder policy label.

priority

Positive integer specifying the priority of the policy. A lower number specifies a higher priority. Must be unique within the label. Policies are evaluated in the order of their priority numbers, and the first policy that matches the request is applied.

Minimum value: 1

Maximum value: 2147483647

gotoPriorityExpression

Expression or other value specifying the next policy to evaluate if the current policy evaluates to TRUE. Specify one of the following values:

- * NEXT - Evaluate the policy with the next higher priority number.
- * END - End policy evaluation.
- * USE_INVOCATION_RESULT - Applicable if this policy invokes another policy label. If the final goto in the invoked policy label has a value of END, the evaluation stops. If the final goto is anything other than END, the current policy label performs a NEXT.
- * A default syntax or classic expression that evaluates to a number.

If you specify an expression, the number to which it evaluates determines the next policy to evaluate, as follows:

- * If the expression evaluates to a higher numbered priority, the policy with that priority is evaluated next.
- * If the expression evaluates to the priority of the current policy, the policy with the next higher numbered priority is evaluated next.
- * If the expression evaluates to a number that is larger than the largest numbered priority, policy evaluation ends.

An UNDEF event is triggered if:

- * The expression is invalid.
- * The expression evaluates to a priority number that is smaller than the current policy's priority number.
- * The expression evaluates to a priority number that is between the current policy's priority number (say, 30) and the highest priority number (say, 100), but does not match any configured priority number (for example, the expression evaluates to the number 85). This example assumes that the priority number increments by 10 for every successive policy, and therefore a priority number of 85 does not exist in the policy label.

invoke

If the current policy evaluates to TRUE, terminate evaluation of policies bound to the current policy label and evaluate the specified policy label.

labelType

Type of policy label to invoke. Available settings function as follows:

- * vserver - Invoke an unnamed policy label associated with a virtual server.
- * policylabel - Invoke a user-defined policy label.

Possible values: vserver, policylabel

Example

```
i) bind responder policylabel resp_lab pol_resp 1 2 ii) bind responder policylabel resp_
```

unbind responder policylabel

Unbinds the specified responder policy from the specified policy label.

Synopsis

```
unbind responder policylabel <labelName> <policyName> [-priority <positive_integer>]
```

Arguments

labelName

Name for the responder policy label. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) hash (#), space (), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after the responder policy label is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my responder policy label" or my responder policy label').

policyName

The name of the policy to be unbound.

priority

Priority of the NOPOLICY to be unbound.

Minimum value: 1

Maximum value: 2147483647

Example

```
unbind responder policylabel resp_lab pol_resp
```

show responder policylabel

Displays the current settings for the specified responder policy label. If no policy label is specified, displays a list of all responder policy labels currently configured on the NetScaler appliance, with abbreviated settings.

Synopsys

show responder policylabel [<labelName>]

Arguments

labelName

Name of the responder policy label.

Outputs

policylabeltype

The type of the policy label.

stateflag

numpol

number of policies bound to label.

hits

Number of times policy label was invoked.

policyName

Name of the responder policy.

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

invoke

If the current policy evaluates to TRUE, terminate evaluation of policies bound to the current policy label and evaluate the specified policy label.

labelType

Type of policy label to invoke. Available settings function as follows:

- * vserver - Invoke an unnamed policy label associated with a virtual server.
- * policylabel - Invoke a user-defined policy label.

labelName

- * If labelType is policylabel, name of the policy label to invoke.
- * If labelType is reqvserver or resvserver, name of the virtual server.

flags

comment

Any comments to preserve information about this responder policy label.

devno

count

Example

```
i) show responder policylabel resp_lab ii) show responder policylabel
```

stat responder policylabel

Displays statistics for the specified responder policy label. If no policy label name is provided, displays abbreviated statistics for all responder policy labels currently configured on the NetScaler appliance.

Synopsys

```
stat responder policylabel [<labelName>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile  
<input_filename>] [-clearstats ( basic | full )]
```

Arguments

labelName

Name of the responder policy label.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Policy Label Hits (Hits)

Number of times policy label was invoked.

rename responder policylabel

Renames the specified responder policy label.

Synopsys

```
rename responder policylabel <labelName>@ <newName>@
```

Arguments

labelName

Current name of the responder policy label.

newName

New name for the responder policy label. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) hash (#), space (), at (@), equals (=), colon (:), and underscore characters.

Example

```
rename responder policylabel oldname newname
```

Rewrite Commands

The entities on which you can perform NetScaler CLI operations:

- `rewrite action`
- `rewrite global`
- `rewrite param`
- `rewrite policy`
- `rewrite policylabel`

rewrite action

The following operations can be performed on "rewrite action":

add | **rm** | **set** | **unset** | **show** | **rename**

add rewrite action

Creates a rewrite action, which specifies exactly what modifications to make to a request or response before forwarding that request or response to the protected web server or to the user. In addition to user-defined actions, the rewrite feature has the following three built-in actions: * NOREWRITE - Sends the request or response to the user without rewriting it. * RESET - Resets the connection and notifies the user's browser, so that the user can resend the request. * DROP - Drops the connection without sending a response to the user. One of the following three flow types is implicitly associated with every action: * Request - Action applies to the request. * Response - Action applies to the response. * Neutral - Action applies to both requests and responses.

Synopsys

```
add rewrite action <name> <type> <target> [<stringBuilderExpr>] [-pattern <expression> | -search <expression>] [-bypassSafetyCheck ( YES | NO )] [-refineSearch <string>] [-comment <string>]
```

Arguments

name

Name for the user-defined rewrite action. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) hash (#), space (), at (@), equals (=), colon (:), and underscore characters. Can be changed after the rewrite policy is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my rewrite action" or ?my rewrite action?).

type

Type of user-defined rewrite action. The information that you provide for, and the effect of, each type are as follows::

- * REPLACE <target> <string_builder_expr>. Replaces the string with the string-builder expression.
- * REPLACE_ALL <target> <string_builder_expr1> -(pattern|search) <string_builder_expr2>. In the request or response specified by <target>, replaces all occurrences of the string defined by <string_builder_expr1> with the string defined by <string_builder_expr2>. You can use a PCRE-format pattern or the search facility to find the strings to be replaced.
- * REPLACE_HTTP_RES <string_builder_expr>. Replaces the complete HTTP response with the string defined by the string-builder expression.
- * REPLACE_SIP_RES <target> - Replaces the complete SIP response with the string specified by <target>.
- * INSERT_HTTP_HEADER <header_string_builder_expr> <contents_string_builder_expr>. Inserts the HTTP header specified by <header_string_builder_expr> and header contents specified by <contents_string_builder_expr>.
- * DELETE_HTTP_HEADER <target>. Deletes the HTTP header specified by <target>.
- * CORRUPT_HTTP_HEADER <target>. Replaces the header name of all occurrences of the HTTP header specified by <target> with a corrupted name, so that it will not be recognized by the receiver Example: MY_HEADER is changed to MHEY_ADNER.
- * INSERT_BEFORE <string_builder_expr1> <string_builder_expr1>. Finds the string specified in <string_builder_expr1> and inserts the string in <string_builder_expr2> before it.
- * INSERT_BEFORE_ALL <target> <string_builder_expr1> -(pattern|search) <string_builder_expr2>. In the request or response specified by <target>, locates all occurrences of the string specified in <string_builder_expr1> and inserts the string specified in <string_builder_expr2> before each. You can use a PCRE-format pattern or the search facility to find the strings.

* **INSERT_AFTER** <string_builder_expr1> <string_builder_expr2>. Finds the string specified in <string_builder_expr1>, and inserts the string specified in <string_builder_expr2> after it.

* **INSERT_AFTER_ALL** <target> <string_builder_expr1> -(pattern|search) <string_builder_expr>. In the request or response specified by <target>, locates all occurrences of the string specified by <string_builder_expr1> and inserts the string specified by <string_builder_expr2> after each. You can use a PCRE-format pattern or the search facility to find the strings.

* **DELETE** <target>. Finds and deletes the specified target.

* **DELETE_ALL** <target> -(pattern|search) <string_builder_expr>. In the request or response specified by <target>, locates and deletes all occurrences of the string specified by <string_builder_expr>. You can use a PCRE-format pattern or the search facility to find the strings.

* **REPLACE_DIAMETER_HEADER_FIELD** <target> <field value>. In the request or response modify the header field specified by <target>. Use `Diameter.req.flags.SET(<flag>)` or `Diameter.req.flags.UNSET<flag>` as 'stringbuilderexpression' to set or unset flags.

Possible values: noop, delete, insert_http_header, delete_http_header, corrupt_http_header, insert_before, insert_after, replace, replace_http_res, delete_all, replace_all, insert_before_all, insert_after_all, clientless_vpn_encode, clientless_vpn_encode_all, clientless_vpn_decode, clientless_vpn_decode_all, insert_sip_header, delete_sip_header, corrupt_sip_header, replace_sip_res, replace_diameter_header_field

target

Default syntax expression that specifies which part of the request or response to rewrite.

stringBuilderExpr

Default syntax expression that specifies the content to insert into the request or response at the specified location, or that replaces the specified string.

pattern

Pattern that is used to match multiple strings in the request or response. The pattern may be a string literal (without quotes) or a PCRE-format regular expression with a delimiter that consists of any printable ASCII non-alphanumeric character except for the underscore (`_`) and space () that is not otherwise used in the expression. Example: `re~https?://|HTTPS?://~` The preceding regular expression can use the tilde (`~`) as the delimiter because that character does not appear in the regular expression itself. Used in the **INSERT_BEFORE_ALL**, **INSERT_AFTER_ALL**, **REPLACE_ALL**, and **DELETE_ALL** action types.

search

Search facility that is used to match multiple strings in the request or response. Used in the **INSERT_BEFORE_ALL**, **INSERT_AFTER_ALL**, **REPLACE_ALL**, and **DELETE_ALL** action types. The following search types are supported:

* **Text** ("text(string)") - A literal string. Example: `-search text("hello")`

* **Regular expression** (?regex(re<delimiter>regular exp<delimiter>?)) - Pattern that is used to match multiple strings in the request or response. The pattern may be a string literal (without quotes) or a PCRE-format regular expression with a delimiter that consists of any printable ASCII non-alphanumeric character except for the underscore (`_`) and space () that is not otherwise used in the expression. Example: `-search regex (re~^hello~)` The preceding regular expression can use the tilde (`~`) as the delimiter because that character does not appear in the regular expression itself.

* **XPath** ("xpath(xp<delimiter>xpath expression<delimiter>)") - An XPath expression. Example: `-search xpath (xp%/a/b%)`

* **JSON** ("xpath_json(xp<delimiter>xpath expression<delimiter>)") - An XPath JSON expression. Example: `-search xpath_json(xp%/a/b%)`

NOTE: JSON searches use the same syntax as XPath searches, but operate on JSON files instead of standard XML files.

* **Patset** ("patset(patset)") - A predefined pattern set. Example: `-search patset("patset1")`.

* **Datset** ("dataset(dataset)") - A predefined dataset. Example: `-search dataset("dataset1")`.

* AVP ("avp(avp number)") - AVP number that is used to match multiple AVPs in a Diameter Message.
Example: -search avp(999)

bypassSafetyCheck

Bypass the safety check and allow unsafe expressions. An unsafe expression is one that contains references to message elements that might not be present in all messages. If an expression refers to a missing request element, an empty string is used instead.

Possible values: YES, NO

Default value: NO

refineSearch

Specify additional criteria to refine the results of the search.

Always starts with the "extend(m,n)" operation, where 'm' specifies number of bytes to the left of selected data and 'n' specifies number of bytes to the right of selected data.

You can use refineSearch only on body expressions, and for the INSERT_BEFORE_ALL, INSERT_AFTER_ALL, REPLACE_ALL, and DELETE_ALL action types.

comment

Comment. Can be used to preserve information about this rewrite action.

Example

```
i) add rewrite action act_insert INSERT_HTTP_HEADER change_req "\\\"no change\\\"" . Th:
```

rm rewrite action

Removes a rewrite action.

Synopsis

```
rm rewrite action <name>
```

Arguments

name

Name of the rewrite action to remove.

Example

```
rm rewrite action act_before
```

set rewrite action

Modifies the specified parameters of a rewrite action.

Synopsis

```
set rewrite action <name> [-target <string>] [-stringBuilderExpr <string>] [-pattern <expression> | -search <expression>] [-bypassSafetyCheck ( YES | NO )] [-refineSearch <string>] [-comment <string>]
```

Arguments

name

Name of the rewrite action to modify.

target

Expression that specifies which part of the connection to rewrite.

stringBuilderExpr

Default syntax expression that specifies the content to insert into the request or response at the specified location, or that replaces the specified string.

pattern

Pattern that is used to match multiple strings in the request or response. The pattern may be a string literal (without quotes) or a PCRE-format regular expression with a delimiter that consists of any printable ASCII non-alphanumeric character except for the underscore (`_`) and space () that is not otherwise used in the expression. Example: `re~https?://|HTTPS?://~` The preceding regular expression can use the tilde (`~`) as the delimiter because that character does not appear in the regular expression itself. Used in the `INSERT_BEFORE_ALL`, `INSERT_AFTER_ALL`, `REPLACE_ALL`, and `DELETE_ALL` action types.

search

Search facility that is used to match multiple strings in the request or response. Used in the `INSERT_BEFORE_ALL`, `INSERT_AFTER_ALL`, `REPLACE_ALL`, and `DELETE_ALL` action types. The following search types are supported:

* Text (`"text(string)"`) - A literal string. Example: `-search text("hello")`

* Regular expression (`?regex(re<delimiter>regular exp<delimiter>?)`) - Pattern that is used to match multiple strings in the request or response. The pattern may be a string literal (without quotes) or a PCRE-format regular expression with a delimiter that consists of any printable ASCII non-alphanumeric character except for the underscore (`_`) and space () that is not otherwise used in the expression. Example: `-search regex (re~^hello~)` The preceding regular expression can use the tilde (`~`) as the delimiter because that character does not appear in the regular expression itself.

* XPath (`"xpath(xp<delimiter>xpath expression<delimiter>)"`) - An XPath expression. Example: `-search xpath (xp%/a/b%)`

* JSON (`"xpath_json(xp<delimiter>xpath expression<delimiter>)"`) - An XPath JSON expression. Example: `-search xpath_json(xp%/a/b%)`

NOTE: JSON searches use the same syntax as XPath searches, but operate on JSON files instead of standard XML files.

* Patset (`"patset(patset)"`) - A predefined pattern set. Example: `-search patset("patset1")`.

* Datset (`"dataset(dataset)"`) - A predefined dataset. Example: `-search dataset("dataset1")`.

* AVP (`"avp(avp number)"`) - AVP number that is used to match multiple AVPs in a Diameter Message. Example: `-search avp(999)`

bypassSafetyCheck

Bypass the safety check and allow unsafe expressions. An unsafe expression is one that contains references to message elements that might not be present in all messages. If an expression refers to a missing request element, an empty string is used instead.

Possible values: YES, NO

Default value: NO

refineSearch

Specify additional criteria to refine the results of the search.

Always starts with the `"extend(m,n)"` operation, where 'm' specifies number of bytes to the left of selected data and 'n' specifies number of bytes to the right of selected data.

You can use `refineSearch` only on body expressions, and for the `INSERT_BEFORE_ALL`, `INSERT_AFTER_ALL`, `REPLACE_ALL`, and `DELETE_ALL` action types.

comment

Comment. Can be used to preserve information about this rewrite action.

Example

```
set rewrite action rwact1 -target "HTTP.REQ.HEADER(\\\\\\"MyHdr\\\\\\")" -stringBuilderExpr "I
```

unset rewrite action

Use this command to remove rewrite action settings. Refer to the set rewrite action command for meanings of the arguments.

Synopsys

```
unset rewrite action <name> [-stringBuilderExpr] [-refineSearch] [-comment]
```

show rewrite action

Displays the current settings for the specified rewrite action. If no rewrite action name is provided, displays a list of all rewrite actions currently configured on the NetScaler appliance.

Synopsys

```
show rewrite action [<name>]
```

Arguments

name

Name of the rewrite action.

Outputs

stateflag

type

Type of rewrite action. It can be:
(delete|replace|insert_http_header|insert_before|insert_after|replace_http_res).

target

Expression specifying which part of HTTP header needs to be rewritten.

stringBuilderExpr

Expression specifying the value of rewritten HTTP header.

pattern

Pattern used for insert_before_all, insert_after_all, replace_all, delete_all action types.

search

Search facility that is used to match multiple strings in the request or response. Used in the INSERT_BEFORE_ALL, INSERT_AFTER_ALL, REPLACE_ALL, and DELETE_ALL action types. The following search types are supported:

* Text ("text(string)") - A literal string. Example: -search text("hello")

* Regular expression (?regex(re<delimiter>regular exp<delimiter>?)) - Pattern that is used to match multiple strings in the request or response. The pattern may be a string literal (without quotes) or a PCRE-format regular expression with a delimiter that consists of any printable ASCII non-alphanumeric character except for the underscore (_) and space () that is not otherwise used in the expression. Example: -search regex (re~^hello~) The preceding regular expression can use the tilde (~) as the delimiter because that character does not appear in the regular expression itself.

* XPath ("xpath(xpath<delimiter>xpath expression<delimiter>") - An XPath expression. Example: -search xpath (xp%/a/b%)

* JSON ("xpath_json(xpath<delimiter>xpath expression<delimiter>") - An XPath JSON expression. Example: -search xpath_json(xpath%/a/b%)

NOTE: JSON searches use the same syntax as XPath searches, but operate on JSON files instead of standard XML files.

* Patset ("patset(patset)") - A predefined pattern set. Example: -search patset("patset1").

* Dataset ("dataset(dataset)") - A predefined dataset. Example: -search dataset("dataset1").

* AVP ("avp(avp number)") - AVP number that is used to match multiple AVPs in a Diameter Message. Example: -search avp(999)

bypassSafetyCheck

The safety check to allow unsafe expressions.

refineSearch

Specify additional criteria to refine the results of the search.

Always starts with the "extend(m,n)" operation, where 'm' specifies number of bytes to the left of selected data and 'n' specifies number of bytes to the right of selected data.

You can use refineSearch only on body expressions, and for the INSERT_BEFORE_ALL, INSERT_AFTER_ALL, REPLACE_ALL, and DELETE_ALL action types.

hits

The number of times the action has been taken.

undefHits

The number of times the action resulted in UNDEF.

referenceCount

The number of references to the action.

description

Description of the action

flags

isDefault

A value of true is returned if it is a default rewriteaction.

comment

Comment. Can be used to preserve information about this rewrite action.

builtin

Flag to determine whether rewrite action is built-in or not

devno

count

Example

```
1. show rewrite action 2. show rewrite action act_insert
```

rename rewrite action

Renames a rewrite action.

Synopsys

```
rename rewrite action <name>@ <newName>@
```

Arguments

name

Existing name of the rewrite action.

newName

New name for the rewrite action.

Must begin with a letter, number, or the underscore character (`_`), and must contain only letters, numbers, and the hyphen (`-`), period (`.`) hash (`#`), space (), at (`@`), equals (`=`), colon (`:`), and underscore characters. Can be changed after the rewrite policy is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my rewrite action" or ?my rewrite action?).

Example

```
rename rewrite action oldname newname
```

rewrite global

The following operations can be performed on "rewrite global":

[bind](#) | [unbind](#) | [show](#)

bind rewrite global

Activates the specified rewrite policy globally.

Synopsys

```
bind rewrite global <policyName> <priority> [<gotoPriorityExpression>] [-type <type>] [-invoke (<labelType> <labelName>)]
```

Arguments

policyName

Name of the rewrite policy to activate.

priority

Positive integer specifying the priority of the policy. A lower number specifies a higher priority. Must be unique within the label. Policies are evaluated in the order of their priorities, and the first policy that matches the request or response is applied.

Minimum value: 1

Maximum value: 2147483647

gotoPriorityExpression

Expression or other value specifying the next policy to evaluate if the current policy evaluates to TRUE. Specify one of the following values:

- * NEXT - Evaluate the policy with the next higher priority number.
- * END - End policy evaluation.
- * USE_INVOCATION_RESULT - Applicable if this policy invokes another policy label. If the final goto in the invoked policy label has a value of END, the evaluation stops. If the final goto is anything other than END, the current policy label performs a NEXT.
- * A default syntax or classic expression that evaluates to a number.

If you specify an expression, the number to which it evaluates determines the next policy to evaluate, as follows:

- * If the expression evaluates to a higher numbered priority, the policy with that priority is evaluated next.
- * If the expression evaluates to the priority of the current policy, the policy with the next higher numbered priority is evaluated next.
- * If the expression evaluates to a number that is larger than the largest numbered priority, policy evaluation ends.

An UNDEF event is triggered if:

- * The expression is invalid.
- * The expression evaluates to a priority number that is smaller than the current policy's priority number.
- * The expression evaluates to a priority number that is between the current policy's priority number (say, 30) and the highest priority number (say, 100), but does not match any configured priority number (for example, the expression evaluates to the number 85). This example assumes that the priority number increments by 10 for every successive policy, and therefore a priority number of 85 does not exist in the policy label.

type

Bind point to which to bind the policy. Available settings function as follows:

- * REQ_OVERRIDE - Request override. Binds the policy to the priority request queue.
- * REQ_DEFAULT - Binds the policy to the default request queue.
- * RES_OVERRIDE - Response override. Binds the policy to the priority response queue.
- RES_DEFAULT - Binds the policy to the default response queue.
- OTHERTCP_REQ_OVERRIDE - Binds the policy to the non-HTTP TCP priority request queue.
- OTHERTCP_REQ_DEFAULT - Binds the policy to the non-HTTP TCP default request queue.
- OTHERTCP_RES_OVERRIDE - Binds the policy to the non-HTTP TCP priority response queue.
- OTHERTCP_RES_DEFAULT - Binds the policy to the non-HTTP TCP default response queue.
- SIPUDP_REQ_OVERRIDE - Binds the policy to the SIP priority request queue.
- SIPUDP_REQ_DEFAULT - Binds the policy to the SIP default request queue.
- SIPUDP_RES_OVERRIDE - Binds the policy to the SIP priority response queue.
- SIPUDP_RES_DEFAULT - Binds the policy to the SIP default response queue.

Possible values: REQ_OVERRIDE, REQ_DEFAULT, RES_OVERRIDE, RES_DEFAULT, OTHERTCP_REQ_OVERRIDE, OTHERTCP_REQ_DEFAULT, OTHERTCP_RES_OVERRIDE, OTHERTCP_RES_DEFAULT, SIPUDP_REQ_OVERRIDE, SIPUDP_REQ_DEFAULT, SIPUDP_RES_OVERRIDE, SIPUDP_RES_DEFAULT, DIAMETER_REQ_OVERRIDE, DIAMETER_REQ_DEFAULT, DIAMETER_RES_OVERRIDE, DIAMETER_RES_DEFAULT

invoke

Terminate evaluation of policies bound to the current policy label, and then forward the request to the specified virtual server or evaluate the specified policy label.

labelType

Type of invocation. Available settings function as follows:

- * reqserver - Forward the request to the specified request virtual server.
- * resvserver - Forward the response to the specified response virtual server.
- * policylabel - Invoke the specified policy label.

Possible values: reqserver, resvserver, policylabel

labelName

- * If labelType is policylabel, name of the policy label to invoke.
- * If labelType is reqserver or resvserver, name of the virtual server to which to forward the request of response.

Example

```
i) bind rewrite global pol9 9   ii) bind rewrite global pol9 9 120   iii) bind rewrite glo
```

unbind rewrite global

Unbinds the specified rewrite policy from rewrite global. See the bind rewrite global command for a description of the parameters.

Synopsys

unbind rewrite global <policyName> [-type <type>] [-priority <positive_integer>]

Arguments

policyName

Name of the rewrite policy to deactivate.

type

The bindpoint from which to unbind.

Possible values: REQ_OVERRIDE, REQ_DEFAULT, RES_OVERRIDE, RES_DEFAULT, OTHERTCP_REQ_OVERRIDE, OTHERTCP_REQ_DEFAULT, OTHERTCP_RES_OVERRIDE, OTHERTCP_RES_DEFAULT, SIPUDP_REQ_OVERRIDE, SIPUDP_REQ_DEFAULT, SIPUDP_RES_OVERRIDE, SIPUDP_RES_DEFAULT, DIAMETER_REQ_OVERRIDE, DIAMETER_REQ_DEFAULT, DIAMETER_RES_OVERRIDE, DIAMETER_RES_DEFAULT

priority

Priority of the NOPOLICY to be unbound.

Minimum value: 1

Maximum value: 2147483647

Example

```
unbind rewrite global pol9
```

show rewrite global

Displays the list of policies bound to the specified rewrite global policy bank. If no policy bank is specified, displays a list of all policies bound to rewrite global.

Synopsys

show rewrite global [-type <type>]

Arguments

type

The bindpoint to which to policy is bound.

Possible values: REQ_OVERRIDE, REQ_DEFAULT, RES_OVERRIDE, RES_DEFAULT, OTHERTCP_REQ_OVERRIDE, OTHERTCP_REQ_DEFAULT, OTHERTCP_RES_OVERRIDE, OTHERTCP_RES_DEFAULT, SIPUDP_REQ_OVERRIDE, SIPUDP_REQ_DEFAULT, SIPUDP_RES_OVERRIDE, SIPUDP_RES_DEFAULT, DIAMETER_REQ_OVERRIDE, DIAMETER_REQ_DEFAULT, DIAMETER_RES_OVERRIDE, DIAMETER_RES_DEFAULT

Outputs

stateflag

policyName

Name of the rewrite policy.

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

invoke

Terminate evaluation of policies bound to the current policy label, and then forward the request to the specified virtual server or evaluate the specified policy label.

labelType

Type of invocation. Available settings function as follows:

- * reqvserver - Forward the request to the specified request virtual server.
- * resvserver - Forward the response to the specified response virtual server.
- * policylabel - Invoke the specified policy label.

labelName

- * If labelType is policylabel, name of the policy label to invoke.
- * If labelType is reqvserver or resvserver, name of the virtual server to which to forward the request of response.

numpol

The number of policies bound to the bindpoint.

flowType

flowtype of the bound rewrite policy.

flags**devno****count**

Example

```
show rewrite global
```

rewrite param

The following operations can be performed on "rewrite param":

[set](#) | [unset](#) | [show](#)

set rewrite param

Sets the default rewrite undefined action. If an UNDEF event is triggered during policy evaluation and if no undefAction is specified for the current policy, this value is used.

Synopsys

```
set rewrite param -undefAction <string>
```

Arguments

undefAction

Action to perform if the result of policy evaluation is undefined (UNDEF). An UNDEF event indicates an internal error condition.

Available settings function as follows:

- * NOOP - Send the request to the protected server instead of responding to it.
- * RESET - Reset the request and notify the user's browser, so that the user can resend the request.
- * DROP - Drop the request without sending a response to the user.

Default value: "NOREWRITE"

Example

```
set rewrite param -undefAction RESET
```

unset rewrite param

Resets the global undefAction to NOREWRITE..Refer to the set rewrite param command for meanings of the arguments.

Synopsys

```
unset rewrite param -undefAction
```

Example

```
unset rewrite param -undefAction
```

show rewrite param

Displays the default rewrite undefAction.

Synopsys

```
show rewrite param
```

Outputs

undefAction

Name of the rewrite action.

Example

```
show rewrite param
```

rewrite policy

The following operations can be performed on "rewrite policy":

add | **rm** | **set** | **unset** | **show** | **stat** | **rename**

add rewrite policy

Creates a rewrite policy, which specifies which requests or responses to rewrite.

Synopsys

```
add rewrite policy <name> <rule> <action> [<undefAction>] [-comment <string>] [-logAction <string>]
```

Arguments

name

Name for the rewrite policy. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) hash (#), space (), at (@), equals (=), colon (:), and underscore characters. Can be changed after the rewrite policy is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my rewrite policy" or ?my rewrite policy?).

rule

Expression against which traffic is evaluated. Written in default syntax.

Note:

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

(Classic expressions are not supported in the cluster build.)

The following requirements apply only to the NetScaler CLI:

- * If the expression includes one or more spaces, enclose the entire expression in double quotation marks.
- * If the expression itself includes double quotation marks, escape the quotations by using the \ character.
- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Name of the rewrite action to perform if the request or response matches this rewrite policy.

There are also some built-in actions which can be used. These are:

- * NOREWRITE - Send the request from the client to the server or response from the server to the client without making any changes in the message.
- * RESET - Resets the client connection by closing it. The client program, such as a browser, will handle this and may inform the user. The client may then resend the request if desired.
- * DROP - Drop the request without sending a response to the user.

undefAction

Action to perform if the result of policy evaluation is undefined (UNDEF). An UNDEF event indicates an internal error condition. Only the above built-in actions can be used.

comment

Any comments to preserve information about this rewrite policy.

logAction

Name of message log action to use when a request matches this policy.

Example

```
i) add rewrite policy pol9 "HTTP.REQ.HEADER(\\\\"header\\\\" ).CONTAINS(\\\\"qh3\\\\" )" ac
```

rm rewrite policy

Removes the specified rewrite policy.

Synopsis

```
rm rewrite policy <name>
```

Arguments

name

Name of the rewrite policy to be removed.

Example

```
rm rewrite policy pol9
```

set rewrite policy

Modifies the specified parameters of a rewrite policy.

Synopsis

```
set rewrite policy <name> [-rule <expression>] [-action <string>] [-undefAction <string>] [-comment <string>] [-logAction <string>]
```

Arguments

name

Name of the rewrite policy to modify.

rule

Expression against which traffic is evaluated. Written in default syntax.

Note:

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

(Classic expressions are not supported in the cluster build.)

The following requirements apply only to the NetScaler CLI:

- * If the expression includes one or more spaces, enclose the entire expression in double quotation marks.
- * If the expression itself includes double quotation marks, escape the quotations by using the \ character.
- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Name of the rewrite action to perform if the request or response matches this rewrite policy.

There are also some built-in actions which can be used. These are:

- * NOREWRITE - Send the request from the client to the server or response from the server to the client without making any changes in the message.
- * RESET - Resets the client connection by closing it. The client program, such as a browser, will handle this and may inform the user. The client may then resend the request if desired.
- * DROP - Drop the request without sending a response to the user.

undefAction

Action to perform if the result of policy evaluation is undefined (UNDEF). An UNDEF event indicates an internal error condition. Only the above built-in actions can be used.

comment

Any comments to preserve information about this rewrite policy.

logAction

Name of message log action to use when a request matches this policy.

Example

```
set rewrite policy pol9 -rule "HTTP.REQ.HEADER(\\\\\\"header\\\\\\").CONTAINS(\\\\\\"qh2\\\\\\")"
```

unset rewrite policy

Removes the settings of an existing rewrite policy. Attributes for which a default value is available revert to their default values. See the set rewrite policy command for a description of the parameters..Refer to the set rewrite policy command for meanings of the arguments.

Synopsys

```
unset rewrite policy <name> [-undefAction] [-comment] [-logAction]
```

Example

```
unset rewrite policy pol9 -undefAction
```

show rewrite policy

Displays the current settings for the specified rewrite policy. If no policy name is provided, displays a list of all rewrite policies currently configured on the NetScaler appliance.

Synopsys

```
show rewrite policy [<name>] show rewrite policy stats - alias for 'stat rewrite policy'
```

Arguments

name

Name of the rewrite policy.

Outputs

stateflag

rule

Expression against which traffic is evaluated. Written in default syntax.

Note:

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

(Classic expressions are not supported in the cluster build.)

The following requirements apply only to the NetScaler CLI:

- * If the expression includes one or more spaces, enclose the entire expression in double quotation marks.
- * If the expression itself includes double quotation marks, escape the quotations by using the \ character.
- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Rewrite action associated with the policy.

undefAction

Undef Action associated with the policy.

hits

Number of hits.

undefHits

Number of Undef hits.

activePolicy

Indicates whether policy is bound or not.

boundTo

Location where policy is bound

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

labelType

Type of policy label invocation.

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

description

Description of the policy

comment

Any comments to preserve information about this rewrite policy.

logAction

Name of message log action to use when a request matches this policy.

bindPolicyType

isDefault

A value of true is returned if it is a default rewritepolicy.

vserverType

builtin

Flag to determine if rewrite policy is built-in or not

devno

count

Example

```
show rewrite policy
```

stat rewrite policy

Displays statistics for the specified rewrite policy. If no policy name is specified, displays abbreviated statistics for all rewrite policies currently configured on the NetScaler appliance.

Synopsys

```
stat rewrite policy [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats  
( basic | full )]
```

Arguments

name

Name of the rewrite policy.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Policy hits (Hits)

Number of hits on the policy

Policy undef hits (Undefhits)

Number of undef hits on the policy

Example

```
stat rewrite policy
```

rename rewrite policy

Renames the specified rewrite policy. You must restart the NetScaler appliance to put new name in effect.

Synopsys

```
rename rewrite policy <name>@ <newName>@
```

Arguments

name

Existing name of the rewrite policy.

newName

New name for the rewrite policy.

Must begin with a letter, number, or the underscore character (), and must contain only letters, numbers, and the hyphen (-), period (.) hash (#), space (), at (@), equals (=), colon (:), and underscore characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my rewrite policy" or ?my rewrite policy?).

Example

```
rename rewrite policy oldname newname
```

rewrite policylabel

The following operations can be performed on "rewrite policylabel":

add | **rm** | **bind** | **unbind** | **show** | **stat** | **rename**

add rewrite policylabel

Creates a user-defined rewrite policy label.

Synopsys

`add rewrite policylabel <labelName> <transform> [-comment <string>]`

Arguments

labelName

Name for the rewrite policy label. Must begin with a letter, number, or the underscore character (`_`), and must contain only letters, numbers, and the hyphen (`-`), period (`.`) hash (`#`), space (), at (`@`), equals (`=`), colon (`:`), and underscore characters. Cannot be changed after the rewrite policy label is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my rewrite policy label" or ?my rewrite policy label?).

transform

Types of transformations allowed by the policies bound to the label. For Rewrite, the following types are supported:

- * `http_req` - HTTP requests
- * `http_res` - HTTP responses
- * `othertcp_req` - Non-HTTP TCP requests
- * `othertcp_res` - Non-HTTP TCP responses
- * `url` - URLs
- * `text` - Text strings
- * `clientless_vpn_req` - NetScaler clientless VPN requests
- * `clientless_vpn_res` - NetScaler clientless VPN responses
- * `sipudp_req` - SIP requests
- * `sipudp_res` - SIP responses
- * `diameter_req` - DIAMETER requests
- * `diameter_res` - DIAMETER responses

Possible values: `http_req`, `http_res`, `othertcp_req`, `othertcp_res`, `url`, `text`, `clientless_vpn_req`, `clientless_vpn_res`, `sipudp_req`, `sipudp_res`, `diameter_req`, `diameter_res`

comment

Any comments to preserve information about this rewrite policy label.

Example

```
add rewrite policylabel trans_http_url http_req
```

rm rewrite policylabel

Removes the specified rewrite policy label.

Synopsys

```
rm rewrite policylabel <labelName>
```

Arguments

labelName

Name of the rewrite policy label to remove.

Example

```
rm rewrite policylabel trans_http_url
```

bind rewrite policylabel

Binds the specified rewrite policy to the specified policy label.

Synopsys

```
bind rewrite policylabel <labelName> <policyName> <priority> [<gotoPriorityExpression>] [-invoke (<labelType>  
<labelName>)]
```

Arguments

labelName

- * If labelType is policylabel, name of the policy label to invoke.
- * If labelType is reqvserver or resvserver, name of the virtual server to which to forward the request or response.

policyName

Name of the rewrite policy to bind to the policy label.

priority

Positive integer specifying the priority of the policy. A lower number specifies a higher priority. Must be unique within the label. Policies are evaluated in the order of their priorities, and the first policy that matches the request or response is applied.

Minimum value: 1

Maximum value: 2147483647

gotoPriorityExpression

Expression or other value specifying the next policy to evaluate if the current policy evaluates to TRUE. Specify one of the following values:

- * NEXT - Evaluate the policy with the next higher priority number.
- * END - End policy evaluation.
- * USE_INVOCATION_RESULT - Applicable if this policy invokes another policy label. If the final goto in the invoked policy label has a value of END, the evaluation stops. If the final goto is anything other than END, the current policy label performs a NEXT.
- * A default syntax or classic expression that evaluates to a number.

If you specify an expression, the number to which it evaluates determines the next policy to evaluate, as follows:

- * If the expression evaluates to a higher numbered priority, the policy with that priority is evaluated next.
- * If the expression evaluates to the priority of the current policy, the policy with the next higher numbered priority is evaluated next.
- * If the expression evaluates to a number that is larger than the largest numbered priority, policy evaluation ends.

An UNDEF event is triggered if:

- * The expression is invalid.
- * The expression evaluates to a priority number that is smaller than the current policy's priority number.
- * The expression evaluates to a priority number that is between the current policy's priority number (say, 30) and the highest priority number (say, 100), but does not match any configured priority number (for example, the expression evaluates to the number 85). This example assumes that the priority number increments by 10 for every successive policy, and therefore a priority number of 85 does not exist in the policy label.

invoke

Suspend evaluation of policies bound to the current policy label, and then forward the request to the specified virtual server or evaluate the specified policy label.

labelType

Type of invocation. Available settings function as follows:

- * reqvserver - Forward the request to the specified request virtual server.
- * resvserver - Forward the response to the specified response virtual server.
- * policylabel - Invoke the specified policy label.

Possible values: reqvserver, resvserver, policylabel

Example

```
i) bind rewrite policylabel trans_http_url pol_1 1 2 -invoke reqvserver CURRENT ii) bind
```

unbind rewrite policylabel

Unbinds the specified rewrite policy from the specified policy label. See the bind rewrite policylabel command for a description of the parameters.

Synopsys

```
unbind rewrite policylabel <labelName> <policyName> [-priority <positive_integer>]
```

Arguments

labelName

Name for the rewrite policy label. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) hash (#), space (), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after the rewrite policy label is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my rewrite policy label" or ?my rewrite policy label?).

policyName

Name of the rewrite policy to bind to the policy label.

priority

Priority of the NOPOLICY to be unbound.

Minimum value: 1

Maximum value: 2147483647

Example

```
unbind rewrite policylabel trans_http_url pol_1
```

show rewrite policylabel

Displays the current settings for the specified rewrite policy label. If no policy label is specified, displays a list of all rewrite policy labels currently configured on the NetScaler appliance.

Synopsys

```
show rewrite policylabel [<labelName>]
```

Arguments

labelName

Name of the rewrite policy label.

Outputs

stateflag

transform

Types of transformations allowed by the policies bound to the label. For Rewrite, the following types are supported:

- * http_req - HTTP requests
- * http_res - HTTP responses
- * othertcp_req - Non-HTTP TCP requests
- * othertcp_res - Non-HTTP TCP responses
- * url - URLs
- * text - Text strings
- * clientless_vpn_req - NetScaler clientless VPN requests
- * clientless_vpn_res - NetScaler clientless VPN responses
- * sipudp_req - SIP requests
- * sipudp_res - SIP responses
- * diameter_req - DIAMETER requests
- * diameter_res - DIAMETER responses

numpol

Number of policies bound to label.

hits

Number of times policy label was invoked.

policyName

Name of the rewrite policy to bind to the policy label.

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

invoke

Suspend evaluation of policies bound to the current policy label, and then forward the request to the specified virtual server or evaluate the specified policy label.

labelType

Type of invocation. Available settings function as follows:

- * reqvserver - Forward the request to the specified request virtual server.
- * resvserver - Forward the response to the specified response virtual server.
- * policylabel - Invoke the specified policy label.

labelName

- * If labelType is policylabel, name of the policy label to invoke.
- * If labelType is reqvserver or resvserver, name of the virtual server to which to forward the request or response.

flowType

Flowtype of the bound rewrite policy.

description

Description of the policylabel

isDefault

A value of true is returned if it is a default rewritepolicylabel.

flags

comment

Any comments to preserve information about this rewrite policy label.

builtin

Flag to determine if rewrite policy label is built-in or not

devno

count

Example

```
i) show rewrite policylabel trans_http_url ii) show rewrite policylabel
```

stat rewrite policylabel

Displays statistics for the specified rewrite policy label. If no policy label name is provided, displays abbreviated statistics for all rewrite policy labels currently configured on the NetScaler appliance.

Synopsys

stat rewrite policylabel [<labelName>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full)]

Arguments

labelName

Name of the rewrite policy label.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Policy Label Hits (Hits)

Number of times policy label was invoked.

rename rewrite policylabel

Renames a rewrite policy label.

Synopsys

rename rewrite policylabel <labelName>@ <newName>@

Arguments

labelName

Current name of the policy label.

newName

New name for the rewrite policy label.

Must begin with a letter, number, or the underscore character (), and must contain only letters, numbers, and the hyphen (-), period (.) hash (#), space (), at (@), equals (=), colon (:), and underscore characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my policy label" or ?my policy label?).

Example

```
rename rewrite policylabel oldname newname
```

RISE Commands

The entities on which you can perform NetScaler CLI operations:

- o [rise apbrSvc](#)
- o [rise param](#)
- o [rise profile](#)
- o [rise rhi](#)

rise apbrSvc

The following operations can be performed on "rise apbrSvc":

show rise apbrSvc

Retrieves configured APBR services

Synopsys

show rise apbrSvc

Outputs

name

Name for the APBR service

riseSvcType

Service or Service Group

serverIP

Server IP for APBR service

nextHopIP

Nexthop IP for APBR service

vlan

Vlan id for APBR service

protocol

Protocol type for APBR service

serverPort

Server port for APBR service

devno**count****stateflag**

rise param

The following operations can be performed on "rise param":

[set](#) | [unset](#) | [show](#)

set rise param

Sets the global parameters for RISE

Synopsis

```
set rise param [-directMode ( ENABLED | DISABLED )] [-indirectMode ( ENABLED | DISABLED )]
```

Arguments

directMode

RISE Direct attach mode

Possible values: ENABLED, DISABLED

Default value: ENABLED

indirectMode

RISE Indirect attach mode

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
set riseParam -directMode ENABLED
```

unset rise param

Use this command to remove rise param settings. Refer to the set rise param command for meanings of the arguments.

Synopsis

```
unset rise param [-directMode] [-indirectMode]
```

show rise param

Display the global parameters for RISE

Synopsis

```
show rise param
```

Outputs

directMode

RISE Direct attach mode

indirectMode

RISE Indirect attach mode

Example

show riseParam

rise profile

The following operations can be performed on "rise profile":

show rise profile

Retrieves the RISE profile

Synopsys

show rise profile [<profileName>]

Arguments

profileName

Name of the RISE profile

Outputs

serviceName

RISE Service name

deviceId

Device id

slotID

Slot number of the RISE profile

slotNO

Slot number of the RISE profile

vdclid

RISE vdc id

vpclid

RISE vpc id

IPAddress

RISE ip address

Mode

RISE attach mode

status

RISE status

vlan

RISE Vlan id

vlanGroupId

RISE Vlan Group id

vlanCfgStatus

RISE config status

ifnum

RISE Interface number

memberInterface

RISE profile member interfaces

issu

RISE issu status

devno**count****stateflag**

rise rhi

The following operations can be performed on "rise rhi":

show rise rhi

Retrieves RISE RHI rules programmed

Synopsys

show rise rhi

Outputs

IPAddress

(V)IP advertised

prefixLen

Prefix Length

hostRtGw

Gateway for the advertised IP

nextHopVlan

Vlan id on which the gateway is present

weight

Cost of the route

vserverRHILevel

Advertisement level

devno

count

stateflag

Router Commands

The entities on which you can perform NetScaler CLI operations:

- `router bgp`
- `router dynamicRouting`
- `router ospf`
- `router rip`
- `vtysh`

router bgp

The following operations can be performed on "router bgp":

[add](#) | [clear](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add router bgp

NOTE: This command is deprecated. All routing configurations have now been moved to vtysh

Synopsys

Arguments

autonomousSystem

The BGP autonomous system.

Minimum value: 1

routerID

The router ID of the router.

learnRoute

The state of route learning from BGP.

staticRedistribute

The state of router in redistribution of static routes.

staticRouteMap

The route map to apply while redistributing static routes.

kernelRedistribute

The state of router in redistribution of kernel routes.

kernelRouteMap

The route map to apply while redistributing kernel routes.

conRedistribute

The state of router in redistribution of connected routes.

connectedRouteMap

The route map to apply while redistributing connected routes.

neighbor

Add a BGP neighbor.

remoteAS

The AS of the neighbor.

Minimum value: 0

neighborRouteMap

The route map to apply to the neighbor.

network

The neighbor to be advertised.

netmask

The netmask of the neighbor to be advertised.

clear router bgp

NOTE: This command is deprecated. All routing configurations have now been moved to vtysh

Synopsys

Arguments

autonomousSystem

The autonomous system for BGP.

Minimum value: 1

neighbor

The neighbor associated with the connection that needs to be torn down.

all

Reset TCP connections to all neighbors.

rm router bgp

NOTE: This command is deprecated. All routing configurations have now been moved to vtysh

Synopsys

Arguments

autonomousSystem

The autonomous system for BGP.

Minimum value: 1

neighbor

To remove a particular neighbor.

set router bgp

NOTE: This command is deprecated. All routing configurations have now been moved to vtysh

Synopsys

Arguments

autonomousSystem

The autonomous system for BGP.

Minimum value: 1

routerID

The Router ID of this router.

learnRoute

The state of the router in learning routes from BGP. Use this option to enable route learning and installation from BGP.

staticRedistribute

The state of the router in redistributing static routes. Use this option to enable the redistribution of static routes.

staticRouteMap

The route map to apply while redistributing static routes.

kernelRedistribute

The state of the router in redistribution of kernel routes.

kernelRouteMap

The state of the router in redistributing kernel routes. The route map to apply while redistributing kernel routes.

conRedistribute

The state of the router in redistributing connected routes. Use this option to enable the redistribution of connected routes into the BGP domain.

connectedRouteMap

The route map to apply while redistributing connected routes.

neighbor

The IP address of a BGP peer for the router.

remoteAS

The autonomous system of the peer.

Minimum value: 0

neighborRouteMap

The route map to apply to the specified neighbor.

network

The network to be advertised.

netmask

The netmask of the advertised network.

unset router bgp

Use this command to remove router bgp settings. Refer to the set router bgp command for meanings of the arguments.
NOTE: This command is deprecated. All routing configurations have now been moved to vtysh

Synopsys

show router bgp

NOTE: This command is deprecated. All routing configurations have now been moved to vtysh

Synopsys

Arguments

autonomousSystem

The autonomous system for BGP.

Minimum value: 1

bgpOptions

option to show BGP command either neighbors or summary

Possible values: neighbors, summary

routeMap

The BGP route map.

Outputs

devno

count

stateflag

router dynamicRouting

The following operations can be performed on "router dynamicRouting":

[show](#) | [apply](#)

show router dynamicRouting

show dynamic routing config from ZebOS daemons

Synopsis

show router dynamicRouting [-commandString <string>]

Arguments

commandString

command to be executed

Outputs

output

command output

devno

count

stateflag

apply router dynamicRouting

apply dynamic routing to ZebOS daemons

Synopsis

apply router dynamicRouting [-commandString <string>]

Arguments

commandString

command to be executed

router ospf

The following operations can be performed on "router ospf":

[set](#) | [unset](#) | [show](#)

set router ospf

Configure different OSPF parameters. NOTE: This command is deprecated. All routing configurations have now been moved to vtysh

Synopsys

Arguments

routerID

The router ID.

passiveInterface

The mode of the Interface. Use this option to change the mode of the interface to listen only.

staticRedistribute

The state of the router in redistributing static routes. Use this option to enable the redistribution of static routes.

staticMetricType

The metric type for static routes.

Default value: 2

Minimum value: 1

Maximum value: 2

kernelRedistribute

The state of the router in redistributing kernel routes. Use this option to enable the redistribution of kernel routes.

kernelMetricType

The metric type for kernel routes.

Default value: 2

Minimum value: 1

Maximum value: 2

conRedistribute

The state of the router in redistributing connected routes. Use this option to enable the redistribution of connected routes.

conMetricType

The metric type for connected routes.

Default value: 2

Minimum value: 1

Maximum value: 2

learnRoute

The state of the router in learning routes from OSPF. Use this option to enable route learning from OSPF.

network

The broadcast network on which OSPF is to be run.

netmask

The netmask for the broadcast network.

area

The area ID of the area in which OSPF is running.

Default value: -1

host

The stub link.

cost

The cost of the hostroute.

Example

```
set ospf -routerID 1.2.3.4
```

unset router ospf

Unset the OSPF parameters that were configured using the `###set ospf###` command..Refer to the `set router ospf` command for meanings of the arguments.NOTE: This command is deprecated.All routing configurations have now been moved to vtysh

Synopsys

Example

```
unset ospf -router-id
```

show router ospf

Display the state of the OSPF daemon. NOTE: This command is deprecated.All routing configurations have now been moved to vtysh

Synopsys

Arguments

ospfoptions

The Router OSPF option. Use this option to display one of border-routers, database, interface, neighbor, route, and virtual-links.

Possible values: border-routers, database, interface, neighbor, route, virtual-links

Outputs

network

The network on which OSPF is running.

netmask

Netmask of the network on which OSPF is running

Example

```
show ospf neighbor
```

router rip

The following operations can be performed on "router rip":

[set](#) | [unset](#) | [show](#)

set router rip

Configure the RIP daemon. NOTE: This command is deprecated. All routing configurations have now been moved to vtysh

Synopsys

Arguments

defaultMetric

The default metrics when advertising routes.

Default value: 1

Minimum value: 1

Maximum value: 16

passiveInterface

The mode of the interface to listen only.

learnRoute

The state of Route learning. Use this option to enable route learning and installation in the kernel.

staticRedistribute

The state of redistributing static routes.

kernelRedistribute

The state of redistributing kernel routes.

network

The broadcast network on which RIP must run.

netmask

The netmask for the network on which RIP must run.

Example

```
set router rip -kernelRedistribute
```

unset router rip

Unset the RIP parameters..Refer to the set router rip command for meanings of the arguments.NOTE: This command is deprecated. All routing configurations have now been moved to vtysh

Synopsys

Example

```
unset rip -default-metric
```

show router rip

Display the RIP configuration. NOTE: This command is deprecated. All routing configurations have now been moved to vtysh

Synopsys

Arguments

ripOptions

RIP option in show command, one of database or interface.

Possible values: database, interface

Outputs

network

The broadcast network on which RIP must run.

netmask

Example

```
show rip interface
```

vtysh

The following operations can be performed on "vtysh":

vtysh

Enters into the Virtual Teletype Shell (VTYSH) prompt, at which you can configure all the dynamic routing protocols. The NetScaler dynamic routing suite is based on ZebOS, the commercial version of GNU Zebra.

Synopsys

vtysh

SureConnect Commands

The entities on which you can perform NetScaler CLI operations:

- `sc`
- `sc parameter`
- `sc policy`
- `sc stats`

sc

The following operations can be performed on "sc":

stat sc

Displays SureConnect statistics.

Synopsys

```
stat sc [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

SC condition triggered (ScTrigd)

Number of times that SureConnect conditions were triggered.

SC trigger condition failed

Total number of times SureConnect was not triggered because the thresholds conditions failed.

Policy matches

Total number of incoming requests that matched configured sureconnect policies.

SC responses sent

Total number of in-memory java script served which throws the pop-up window.

Reissued requests (ReissReq)

Total number of reissued SureConnect requests.

Valid reissued requests

Total number of requests that were handled in a single SureConnect session.

Alternate content requests

Total number of alternate content served which throws the pop-up window.

SC POST requests

Total number of HTTP POST requests that triggered SureConnect feature.

SC statistics timeout

Total number of times that SureConnect statistics were reset.

Unsupported browsers

Total number of requests that came from all unsupported browsers.

Tampered SC cookies

Total number of corrupted SureConnect cookies.

sc parameter

The following operations can be performed on "sc parameter":

[set](#) | [unset](#) | [show](#)

set sc parameter

Sets the parameters for displaying SureConnect information.

Synopsis

```
set sc parameter [-sessionLife <secs>] [-vsr <input_filename>]
```

Arguments

sessionLife

Time, in seconds, between the first time and the next time the SureConnect alternative content window is displayed. The alternative content window is displayed only once during a session for the same browser accessing a configured URL, so this parameter determines the length of a session.

Default value: 300

Minimum value: 1

Maximum value: 4294967294

vsr

File containing the customized response to be displayed when the ACTION in the SureConnect policy is set to NS.

Default value: "DEFAULT"

Example

```
set sc parameter -sessionlife 200 -vsr /etc/vsr.htm
```

unset sc parameter

Use this command to remove sc parameter settings. Refer to the set sc parameter command for meanings of the arguments.

Synopsis

```
unset sc parameter [-sessionLife] [-vsr]
```

show sc parameter

Displays the values of the session life and vsr filename parameters.

Synopsis

```
show sc parameter
```

Outputs

sessionLife

The time between first time the Sureconnect alternate content window displays and the next time it displays. The SureConnect alternate content window is displayed only once during a session. For the same browser accessing a configured URL. The value is in seconds.

vsr

The customized response will be displayed to the user if the alternate content server has been determined by the system to have failed.

If you have created a customized response that you want the system to use, enter its filename (if you renamed the vsr.htm file supplied by system). If you have not renamed the file, enter /etc/vsr.htm as the filename.

Example

```
> show sc parameter          Sure Connect Parameters:      Sessionlife: 300      Vsr: DEFAI
```

sc policy

The following operations can be performed on "sc policy":

add | **rm** | **set** | **unset** | **show** | **stat**

add sc policy

Creates a new SureConnect policy.

Synopsys

```
add sc policy <name> [-url <URL> | -rule <expression>] [-delay <usecs>] [-maxConn <positive_integer>] [-action <action> (<altContentSvcName> <altContentPath>)]
```

Arguments

name

Name for the policy. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

url

URL against which to match incoming client request.

rule

Expression against which the traffic is evaluated.

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

The following requirements apply only to the NetScaler CLI:

- * If the expression includes one or more spaces, enclose the entire expression in double quotation marks.
- * If the expression itself includes double quotation marks, escape the quotations by using the \ character.
- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

delay

Delay threshold, in microseconds, for requests that match the policy's URL or rule. If the delay statistics gathered for the matching request exceed the specified delay, SureConnect is triggered for that request.

Minimum value: 1

Maximum value: 599999999

maxConn

Maximum number of concurrent connections that can be open for requests that match the policy's URL or rule.

Minimum value: 1

Maximum value: 4294967294

action

Action to be taken when the delay or maximum-connections threshold is reached. Available settings function as follows:

ACS - Serve content from an alternative content service.

NS - Serve alternative content from the NetScaler appliance.

NO ACTION - Serve no alternative content. However, delay statistics are still collected for the configured URLs, and, if the Maximum Client Connections parameter is set, the number of connections is limited to the value specified by that parameter. (However, alternative content is not served even if the maxConn threshold is met).

Possible values: ACS, NS, NOACTION

altContentSvcName

Name of the alternative content service to be used in the ACS action.

altContentPath

Path to the alternative content service to be used in the ACS action.

Example

```
add sc policy scpol_ns -delay 1000000 -url /delay.asp -action NS add policy expression e:
```

rm sc policy

Removes the specified SureConnect policy.

Synopsys

```
rm sc policy <name>
```

Arguments

name

Name of the policy to be removed.

Example

```
rm sc policy scpol_ns rm sc policy scpol_acs
```

set sc policy

Modifies the specified settings of a SureConnect policy.

Synopsys

```
set sc policy <name> [-url <URL> | -rule <expression>] [-delay <usecs>] [-maxConn <positive_integer>] [-action  
<action> (<altContentSvcName> <altContentPath>)]
```

Arguments

name

Name of the policy to be modified.

url

URL against which to match requests. URLs take precedence over rules in SureConnect policies.

rule

Expression against which the traffic is evaluated.

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

The following requirements apply only to the NetScaler CLI:

- * If the expression includes one or more spaces, enclose the entire expression in double quotation marks.
- * If the expression itself includes double quotation marks, escape the quotations by using the `\` character.
- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

delay

Delay threshold, in microseconds, for requests that match the policy's URL or rule. If the delay statistics gathered for the matching request exceed the specified delay, SureConnect is triggered for that request.

Minimum value: 1

Maximum value: 599999999

maxConn

Maximum number of concurrent connections that can be open for the configured URL or rule.

Minimum value: 1

Maximum value: 4294967294

action

Action to be taken when the delay or maximum-connections threshold is reached. Available settings function as follows:

ACS - Serve content from an alternative content service.

NS - Serve alternative content from the NetScaler appliance.

NO ACTION - Serve no alternative content. However, delay statistics are still collected for the configured URLs, and, if the Maximum Client Connections parameter is set, the number of connections is limited to the value specified by that parameter. (However, alternative content is not served even if the maxConn threshold is met).

Possible values: ACS, NS, NOACTION

altContentSvcName

Name of the alternative content service to be used in the ACS action.

altContentPath

Path to the alternative content service to be used in the ACS action.

Example

```
set sc policy scpol_ns -delay 2000000 set sc policy scpol_acs -maxconn 100
```

unset sc policy

Use this command to remove sc policy settings. Refer to the set sc policy command for meanings of the arguments.

Synopsis

```
unset sc policy <name> [-delay] [-maxConn]
```

show sc policy

Displays information about the SureConnect policies.

Synopsis

show sc policy [<name>]

Arguments

name

Name of a policy about which to display detailed information. To display information about all the SureConnect policies, do not set this parameter.

Outputs

url

The URL name. The system matches the incoming client request against the URL you enter here.

rule

The rule that the system matches with the incoming request.

The system matches the incoming request against the rules you enter here. Before matching against the configured rules, the NetScaler 9000 system matches the requests with any of the configured URLs. Thus, URLs have a higher precedence over rules. If the incoming request does not match any of the configured URLs or the rules that have been configured, then SureConnect does not trigger.

Expression logic is expression names, separated by the logical operators || and && , and possibly grouped using parenthesis. If the expression contains blanks (for example, between an expression name and a logical operator), then the entire argument must be enclosed in double quotes. The following are valid expression logic:

ns_ext_cgi||ns_ext_asp

ns_non_get && (ns_header_cookie||ns_header_pragma)

delay

Delay threshold, in microseconds, for requests that match the policy's URL or rule. If the delay statistics gathered for the matching request exceed the specified delay, SureConnect is triggered for that request.

maxConn

Maximum number of concurrent connections that can be open for requests that match the policy's URL or rule.

action

The action to be taken when the thresholds are met. The valid options are ACS , NS and NOACTION .

ACS - Specifies that alternate content is to be served from altContSvcName with the path altContPath .

NS - Specifies that alternate content is to be served from the NetScaler 9000 system. See the set sc parameter command to customize the response served from the system.

NOACTION - Specifies that no alternate content is to be served. However, delay statistics are still collected for the configured URLs. If the - maxconn argument is specified, the number of connections is limited to that specified value for that configured URL or rule (alternate content will not served even if the - maxconn threshold is met).

altContentSvcName

Name of the alternative content service to be used in the ACS action.

altContentPath

Path to the alternative content service to be used in the ACS action.

devno

count

stateflag

Example

```
> show sc policy          2 monitored Sure Connect Policies: 1)      Name: scpol_ns
```

stat sc policy

Displays statistics about SureConnect policies.

Synopsys

```
stat sc policy [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (
basic | full )]
```

Arguments

name

Name of the policy about which to display statistics. To display statistics about all SureConnect policies, do not set this parameter.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Server TTLB (SvrTTLB)

Server Time-To-Last-Byte in seconds calculated for this SureConnect policy.

Average server TTLB

Average server transaction time in seconds for this SureConnect Policy.

Average client TTLB (AvCITTLB)

Average value of the client Time-To-Last-Byte in seconds for this SureConnect policy.

Physical service IP (SvcIP)

IP address of the service in dotted notation for which these statistics are maintained.

Physical service port (SvcPort)

Port of the service for which these statistics are maintained.

Current client connections (CurClts)

Number of clients currently allowed a server connection by this SureConnect policy.

Current SC queue length (WaitClts)

Current number of SureConnect priority clients that are waiting for a server connection.

Current server connections (CurSvrs)

Current number of open connections to the servers matching this policy.

Estimated waiting time (Sec) (WaitTime)

Value of the currently estimated waiting time in seconds for the configured URL.

Client TCP connections (TotClt)

Total number of clients that were allowed a server connection by this SureConnect policy.

Server TCP connections (TotSvr)

Total number of server connections that were established through this SureConnect policy.

Client HTTP transactions

Total number of client transactions processed by this SureConnect policy.

Server HTTP transactions (SrvTrans)

Number of 200 OK responses received from the web server by this SureConnect policy.

Requests received (TotReq)

Total number of requests received by this SureConnect policy.

Request bytes received (ReqBytes)

Total number of request bytes received by this SureConnect policy.

Server responses received (TotResp)

Total number of server responses received by this SureConnect policy.

Response bytes received (RspBytes)

Total number of response bytes received by this SureConnect policy.

sc stats

The following operations can be performed on "sc stats":

show sc stats

show sc stats is an alias for stat sc

Synopsys

show sc stats - alias for 'stat sc'

SNMP Commands

The entities on which you can perform NetScaler CLI operations:

- o snmp
- o snmp alarm
- o snmp community
- o snmp engineId
- o snmp group
- o snmp manager
- o snmp mib
- o snmp oid
- o snmp option
- o snmp stats
- o snmp trap
- o snmp user
- o snmp view

snmp

The following operations can be performed on "snmp":

stat snmp

Display the statistics related to SNMP.

Synopsys

```
stat snmp [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

SNMP packets received (PktsRx)

SNMP packets received.

SNMP packets sent (PktsTx)

SNMP packets transmitted.

Get requests received (GetReqRx)

SNMP Get-Request PDUs that have been accepted and processed.

Get-next requests received (GtNextRx)

SNMP Get-Next PDUs that have been accepted and processed.

Get-bulk requests received (GtBulkRx)

SNMP Get-Bulk PDUs that have been accepted and processed.

Responses sent (RspTx)

SNMP Get-Response PDUs that have been generated by the NetScaler.

Traps messages sent (TrapsTx)

SNMP Trap PDUs that have been generated by the NetScaler.

Requests dropped (ReqDrop)

SNMP requests dropped.

ASN.1/BER errors in requests (PrsErrRx)

Number of ASN.1 or BER errors encountered when decoding received SNMP Messages.

Unsupported SNMP version (UnkVrsRx)

Number of SNMP messages received, which were for an unsupported SNMP version.

Unknown community name (UnkCNRx)

SNMP messages received, which used an SNMP community name not known to the NetScaler.

No permission on community (BadCURx)

The total number of SNMP Messages received that represented an SNMP operation which was not allowed by the SNMP community named in the Message.

Unsupported security level (UnkSecLv)

SNMP packets that were dropped because they requested a security level that was unknown to the NetScaler or otherwise unavailable.

Not in time window (NtTimeWd)

SNMP packets that were dropped because they appeared outside of the authoritative SNMP engine's window.

Unknown user name (UnkUser)

SNMP packets that were dropped because they referenced a user that was not known to the SNMP engine.

Unknown engine Id (UnkEngId)

SNMP packets that were dropped because they referenced an SNMP engine ID that was not known to the NetScaler.

Wrong digest value (WrgDgst)

SNMP packets that were dropped because they did not contain the expected digest value.

Decryption errors (DcrptErr)

SNMP packets that were dropped because they could not be decrypted.

Example

```
stat snmp
```

snmp alarm

The following operations can be performed on "snmp alarm":

[set](#) | [unset](#) | [enable](#) | [disable](#) | [show](#)

set snmp alarm

Configures an SNMP alarm. You must enable and configure alarms to generate enterprise-specific trap messages. The NetScaler appliance sends these trap messages only to trap listeners of type (class) SPECIFIC. The SNMP alarms are either event based or threshold based. The NetScaler appliance supports the following user configurable alarms: HA-STATE-CHANGE: Change to primary/secondary CPU-USAGE: Individual CPU usage AVERAGE-CPU: Average CPU usage MGMT-CPU: Management CPU usage ENTITY-STATE: Entity state change SYNFLOOD: Global unacknowledged SYN count MEMORY: Memory usage VSERVER-REQRATE: Vserver specific request rate SERVICE-REQRATE: Service specific request rate ENTITY-RXRATE: Entity specific Rx bytes per sec ENTITY-TXRATE: Entity specific Tx bytes per sec ENTITY-SYNFLOOD: Entity specific unacknowledged SYN count CONFIG-CHANGE: System configuration changed SERVICE-MAXCLIENTS: Service hit max-client limit CONFIG-SAVE: System configuration was saved SERVICEGROUP-MEMBER-REQRATE: Request rate on a service group member SERVICEGROUP-MEMBER-MAXCLIENTS: Service group member hits max-client MONITOR-RTO-THRESHOLD: Monitor probe response timeout LOGIN-FAILURE: GUI/CLI/API login failure SSL-CERT-EXPIRY: Certificate expiry FAN-SPEED-LOW: Low fan speed VOLTAGE-LOW: Low voltage VOLTAGE-HIGH: High Voltage TEMPERATURE-HIGH: High temperature CPU-TEMPERATURE-HIGH: High CPU temperature POWER-SUPPLY-FAILURE: Power supply failure DISK-USAGE-HIGH: High disk usage INTERFACE-THROUGHPUT-LOW: Low Interface throughput MON_PROBE_FAILED: Monitor probe failure HA-VERSION-MISMATCH: HA netscaler's OS version mismatch HA-SYNC-FAILURE: HA config synchronization failure HA-NO-HEARTBEATS: No HA hearbeats HA-BAD-SECONDARY-STATE: Secondary state DOWN/UNKNOWN/STAY SECONDARY INTERFACE-BW-USAGE: System aggregate BW usage RATE-LIMIT-THRESHOLD-EXCEEDED: Client exceed rate-limit threshold ENTITY-NAME-CHANGE: Entity name change HA-PROP-FAILURE: HA config propagation failure IP-CONFLICT: IP conflict PF-RL-RATE-THRESHOLD: Platform rate limit in Mbps PF-RL-PPS-THRESHOLD: Platform packets per second limit PF-RL-RATE-PKTS-DROPPED: Packet Drops due to platform rate limit PF-RL-PPS-PKTS-DROPPED: Packet Drops due to platform packet per sec limit APPFW-START-URL: AppFirewall Start URL violation APPFW-DENY-URL: AppFirewall Deny URL violation APPFW-REFERER-HEADER: AppFirewall Referer Header violation APPFW-CSRF-TAG: AppFirewall CSRF Tag violation APPFW-COOKIE: AppFirewall Cookie violation APPFW-FIELD-CONSISTENCY: AppFirewall Field Consistency violation APPFW-BUFFER-OVERFLOW: AppFirewall Buffer Overflow violation APPFW-FIELD-FORMAT: AppFirewall Field Format violation APPFW-SAFE-COMMERCE: AppFirewall Safe Commerce violation APPFW-SAFE-OBJECT: AppFirewall Safe Object violation APPFW-POLICY-HIT: AppFirewall Policy Hit APPFW-VIOLATIONS-TYPE: AppFirewall Content Type violation APPFW-XSS: AppFirewall Cross Site Scripting violation APPFW-XML-XSS: AppFirewall XML Cross Site Scripting violation APPFW-SQL: AppFirewall SQL violation APPFW-XML-SQL: AppFirewall XML SQL violation APPFW-XML-ATTACHMENT: AppFirewall XML Attachment violation APPFW-XML-DOS: AppFirewall XML DoS violation APPFW-XML-VALIDATION: AppFirewall XML Validation violation APPFW-XML-WSI: AppFirewall XML WSI violation APPFW-XML-SCHEMA-COMPILE: AppFirewall XML Schema Compile violation APPFW-XML-SOAP-FAULT: AppFirewall XML Soap Fault violation DNSKEY-EXPIRY: DNSKEY expiry HA-LICENSE-MISMATCH: HA netscaler's license mismatch SSL-CARD-FAILED: SSL Card Failed SSL-CARD-NORMAL: SSL Card Normal WARM-RESTART-EVENT: Warm Restart Event Occurred HARD-DISK-DRIVE-ERRORS: Hard Disk Drive Errors COMPACT-FLASH-ERRORS: Compact Flash Errors CALLHOME-UPLOAD-EVENT: Attempt to upload Show Tech Support Archive 1024KEY-EXCHANGE-RATE: 1024 Key Exchange Rate 2048KEY-EXCHANGE-RATE: 2048 Key Exchange Rate 4096KEY-EXCHANGE-RATE: 4096 Key Exchange Rate SSL-CUR-SESSION-INUSE: SSL Current Sessions In Use CLUSTER-NODE-HEALTH: Cluster Node Health State Change CLUSTER-NODE-QUORUM: Cluster Node View has Quorum CLUSTER-VERSION-MISMATCH: Cluster Node Version Mismatch CLUSTER-CCO-CHANGE: Cluster Configuration Coordinator Change CLUSTER-OVS-CHANGE: Cluster Operational View Set Change CLUSTER-SYNC-FAILURE: Cluster Config Synchronization Failure CLUSTER-PROP-FAILURE: Cluster Config Propagation Failure HA-STICKY-PRIMARY: Fixed primary state owing to max HA flips INBAND-PROTOCOL-VERSION-MISMATCH: Inband protocol mismatch between BR and QoSd SSL-CHIP-REINIT: SSL Chip Reinit VRID-STATE-CHANGE: VRID State Change PORT-ALLOC-FAILED: Port Alloc Failed LLDP-REMOTE-CHANGE: LLDP Remote Change DUPLICATE-IPV6: IPv6 Address got duplicated For the purposes of this command, entity includes vservers and services.

Synopsys

```
set snmp alarm <trapName> [-thresholdValue <positive_integer> [-normalValue <positive_integer>]] [-time <secs>] [-state ( ENABLED | DISABLED )] [-severity <severity>] [-logging ( ENABLED | DISABLED )]
```

Arguments

trapName

Name of the SNMP alarm. This parameter is required for identifying the SNMP alarm and cannot be modified.

Possible values: CPU-USAGE, AVERAGE-CPU, MEMORY, MGMT-CPU-USAGE, SYNFLOOD, VSERVER-REQRATE, SERVICE-REQRATE, ENTITY-RXRATE, ENTITY-TXRATE, ENTITY-SYNFLOOD, SERVICE-MAXCLIENTS, HA-STATE-CHANGE, ENTITY-STATE, CONFIG-CHANGE, CONFIG-SAVE, SERVICEGROUP-MEMBER-REQRATE, SERVICEGROUP-MEMBER-MAXCLIENTS, MONITOR-RTO-THRESHOLD, LOGIN-FAILURE, SSL-CERT-EXPIRY, FAN-SPEED-LOW, VOLTAGE-LOW, VOLTAGE-HIGH, TEMPERATURE-HIGH, CPU-TEMPERATURE-HIGH, POWER-SUPPLY-FAILURE, DISK-USAGE-HIGH, INTERFACE-THROUGHPUT-LOW, MON_PROBE_FAILED, HA-VERSION-MISMATCH, HA-SYNC-FAILURE, HA-NO-HEARTBEATS, HA-BAD-SECONDARY-STATE, INTERFACE-BW-USAGE, RATE-LIMIT-THRESHOLD-EXCEEDED, ENTITY-NAME-CHANGE, HA-PROP-FAILURE, IP-CONFLICT, PF-RL-RATE-THRESHOLD, PF-RL-PPS-THRESHOLD, PF-RL-RATE-PKTS-DROPPED, PF-RL-PPS-PKTS-DROPPED, APPFW-START-URL, APPFW-DENY-URL, APPFW-VIOLATIONS-TYPE, APPFW-REFERER-HEADER, APPFW-CSRF-TAG, APPFW-COOKIE, APPFW-FIELD-CONSISTENCY, APPFW-BUFFER-OVERFLOW, APPFW-FIELD-FORMAT, APPFW-SAFE-COMMERCE, APPFW-SAFE-OBJECT, APPFW-POLICY-HIT, APPFW-XSS, APPFW-XML-XSS, APPFW-SQL, APPFW-XML-SQL, APPFW-XML-ATTACHMENT, APPFW-XML-DOS, APPFW-XML-VALIDATION, APPFW-XML-WSI, APPFW-XML-SCHEMA-COMPILE, APPFW-XML-SOAP-FAULT, DNSKEY-EXPIRY, HA-LICENSE-MISMATCH, SSL-CARD-FAILED, SSL-CARD-NORMAL, WARM-RESTART-EVENT, HARD-DISK-DRIVE-ERRORS, COMPACT-FLASH-ERRORS, CALLHOME-UPLOAD-EVENT, 1024KEY-EXCHANGE-RATE, 2048KEY-EXCHANGE-RATE, 4096KEY-EXCHANGE-RATE, SSL-CUR-SESSION-INUSE, CLUSTER-NODE-HEALTH, CLUSTER-NODE-QUORUM, CLUSTER-VERSION-MISMATCH, CLUSTER-CCO-CHANGE, CLUSTER-OVS-CHANGE, CLUSTER-SYNC-FAILURE, CLUSTER-PROP-FAILURE, HA-STICKY-PRIMARY, INBAND-PROTOCOL-VERSION-MISMATCH, SSL-CHIP-REINIT, VRID-STATE-CHANGE, PORT-ALLOC-FAILED, LLDP-REMOTE-CHANGE, DUPLICATE-IPV6

thresholdValue

Value for the high threshold. The NetScaler appliance generates an SNMP trap message when the value of the attribute associated with the alarm is greater than or equal to the specified high threshold value.

Minimum value: 1

normalValue

Value for the normal threshold. A trap message is generated if the value of the respective attribute falls to or below this value after exceeding the high threshold.

Minimum value: 1

time

Interval, in seconds, at which the NetScaler appliance generates SNMP trap messages when the conditions specified in the SNMP alarm are met. Can be specified for the following alarms: SYNFLOOD, HA-VERSION-MISMATCH, HA-SYNC-FAILURE, HA-NO-HEARTBEATS, HA-BAD-SECONDARY-STATE, CLUSTER-NODE-HEALTH, CLUSTER-NODE-QUORUM, CLUSTER-VERSION-MISMATCH, PORT-ALLOC-FAILED and APPFW traps. Default trap time intervals: SYNFLOOD and APPFW traps = 1sec, PORT-ALLOC-FAILED = 3600sec(1 hour), Other Traps = 86400sec(1 day)

Default value: 1

state

Current state of the SNMP alarm. The NetScaler appliance generates trap messages only for SNMP alarms that are enabled. Some alarms are enabled by default, but you can disable them.

Possible values: ENABLED, DISABLED

Default value: ENABLED

severity

Severity level assigned to trap messages generated by this alarm. The severity levels are, in increasing order of severity, Informational, Warning, Minor, Major, and Critical.

This parameter is useful when you want the NetScaler appliance to send trap messages to a trap listener on the basis of severity level. Trap messages with a severity level lower than the specified level (in the trap listener entry) are not sent.

Possible values: Critical, Major, Minor, Warning, Informational

Default value: Unknown

logging

Logging status of the alarm. When logging is enabled, the NetScaler appliance logs every trap message that is generated for this alarm.

Possible values: ENABLED, DISABLED

Default value: ENABLED

Example

```
set snmp alarm VSERVER-REQRATE -thresholdValue 10000 -normalValue 100
```

unset snmp alarm

Resets the specified parameters of an SNMP alarm to their default settings..Refer to the set snmp alarm command for meanings of the arguments.

Synopsys

```
unset snmp alarm <trapName> [-thresholdValue] [-normalValue] [-time] [-state] [-severity] [-logging]
```

Example

```
unset snmp alarm VSERVER-REQRATE
```

enable snmp alarm

Enables or disables an SNMP alarm. The NetScaler appliance looks for conditions specified in the enabled SNMP alarms. When the condition in any enabled SNMP alarm is met, the appliance generates an SNMP trap message. It does not look for conditions specified in disabled SNMP alarms and therefore does not generate an SNMP trap message when the condition in any disabled SNMP alarm is met. Some alarms are enabled by default, but you can disable them.

Synopsys

```
enable snmp alarm <trapName> ...
```

Arguments

trapName

Name of the SNMP alarm. This parameter is required for identifying the SNMP alarm.

Possible values: CPU-USAGE, AVERAGE-CPU, MEMORY, MGMT-CPU-USAGE, SYNFLOOD, VSERVER-REQRATE, SERVICE-REQRATE, ENTITY-RXRATE, ENTITY-TXRATE, ENTITY-SYNFLOOD, SERVICE-MAXCLIENTS, HA-STATE-CHANGE, ENTITY-STATE, CONFIG-CHANGE, CONFIG-SAVE, SERVICEGROUP-MEMBER-REQRATE, SERVICEGROUP-MEMBER-MAXCLIENTS, MONITOR-RTO-THRESHOLD, LOGIN-FAILURE, SSL-CERT-EXPIRY, FAN-SPEED-LOW, VOLTAGE-LOW, VOLTAGE-HIGH, TEMPERATURE-HIGH, CPU-TEMPERATURE-HIGH, POWER-SUPPLY-FAILURE, DISK-USAGE-HIGH, INTERFACE-THROUGHPUT-LOW, MON_PROBE_FAILED, HA-VERSION-MISMATCH, HA-SYNC-FAILURE, HA-NO-HEARTBEATS, HA-BAD-SECONDARY-STATE, INTERFACE-BW-USAGE, RATE-LIMIT-THRESHOLD-EXCEEDED, ENTITY-NAME-CHANGE, HA-PROP-FAILURE, IP-CONFLICT, PF-RL-RATE-THRESHOLD, PF-RL-PPS-THRESHOLD, PF-RL-RATE-PKTS-DROPPED, PF-RL-PPS-PKTS-DROPPED, APPFW-START-URL, APPFW-DENY-URL, APPFW-VIOLATIONS-TYPE, APPFW-REFERER-HEADER, APPFW-CSRF-TAG, APPFW-COOKIE, APPFW-FIELD-CONSISTENCY, APPFW-BUFFER-OVERFLOW, APPFW-FIELD-FORMAT, APPFW-SAFE-COMMERCE, APPFW-SAFE-OBJECT, APPFW-POLICY-HIT, APPFW-XSS, APPFW-XML-XSS, APPFW-SQL, APPFW-XML-SQL, APPFW-XML-ATTACHMENT, APPFW-XML-DOS, APPFW-XML-VALIDATION, APPFW-XML-WSI, APPFW-XML-SCHEMA-COMPILE, APPFW-XML-SOAP-FAULT, DNSKEY-EXPIRY, HA-LICENSE-MISMATCH, SSL-CARD-FAILED, SSL-CARD-NORMAL, WARM-RESTART-EVENT, HARD-DISK-DRIVE-ERRORS, COMPACT-FLASH-ERRORS, CALLHOME-UPLOAD-EVENT, 1024KEY-EXCHANGE-RATE, 2048KEY-EXCHANGE-RATE, 4096KEY-EXCHANGE-RATE, SSL-CUR-SESSION-INUSE, CLUSTER-NODE-HEALTH, CLUSTER-NODE-QUORUM, CLUSTER-VERSION-MISMATCH, CLUSTER-CCO-CHANGE, CLUSTER-OVS-CHANGE, CLUSTER-SYNC-

FAILURE, CLUSTER-PROP-FAILURE, HA-STICKY-PRIMARY, INBAND-PROTOCOL-VERSION-MISMATCH, SSL-CHIP-REINIT, VRID-STATE-CHANGE, PORT-ALLOC-FAILED, LLDP-REMOTE-CHANGE, DUPLICATE-IPV6

Example

```
enable snmp alarm VSERVER-REQRATE enable snmp alarm CPU SYNFLOOD
```

disable snmp alarm

Disables an SNMP alarm. The NetScaler appliance does not generate trap messages for SNMP alarms that are disabled. Some alarms are enabled by default, but you can disable them.

Synopsys

```
disable snmp alarm <trapName> ...
```

Arguments

trapName

Name of the SNMP alarm. This parameter is required for identifying the SNMP alarm.

Possible values: CPU-USAGE, AVERAGE-CPU, MEMORY, MGMT-CPU-USAGE, SYNFLOOD, VSERVER-REQRATE, SERVICE-REQRATE, ENTITY-RXRATE, ENTITY-TXRATE, ENTITY-SYNFLOOD, SERVICE-MAXCLIENTS, HA-STATE-CHANGE, ENTITY-STATE, CONFIG-CHANGE, CONFIG-SAVE, SERVICEGROUP-MEMBER-REQRATE, SERVICEGROUP-MEMBER-MAXCLIENTS, MONITOR-RTO-THRESHOLD, LOGIN-FAILURE, SSL-CERT-EXPIRY, FAN-SPEED-LOW, VOLTAGE-LOW, VOLTAGE-HIGH, TEMPERATURE-HIGH, CPU-TEMPERATURE-HIGH, POWER-SUPPLY-FAILURE, DISK-USAGE-HIGH, INTERFACE-THROUGHPUT-LOW, MON_PROBE_FAILED, HA-VERSION-MISMATCH, HA-SYNC-FAILURE, HA-NO-HEARTBEATS, HA-BAD-SECONDARY-STATE, INTERFACE-BW-USAGE, RATE-LIMIT-THRESHOLD-EXCEEDED, ENTITY-NAME-CHANGE, HA-PROP-FAILURE, IP-CONFLICT, PF-RL-RATE-THRESHOLD, PF-RL-PPS-THRESHOLD, PF-RL-RATE-PKTS-DROPPED, PF-RL-PPS-PKTS-DROPPED, APPFW-START-URL, APPFW-DENY-URL, APPFW-VIOLATIONS-TYPE, APPFW-REFERER-HEADER, APPFW-CSRF-TAG, APPFW-COOKIE, APPFW-FIELD-CONSISTENCY, APPFW-BUFFER-OVERFLOW, APPFW-FIELD-FORMAT, APPFW-SAFE-COMMERCE, APPFW-SAFE-OBJECT, APPFW-POLICY-HIT, APPFW-XSS, APPFW-XML-XSS, APPFW-SQL, APPFW-XML-SQL, APPFW-XML-ATTACHMENT, APPFW-XML-DOS, APPFW-XML-VALIDATION, APPFW-XML-WSI, APPFW-XML-SCHEMA-COMPILE, APPFW-XML-SOAP-FAULT, DNSKEY-EXPIRY, HA-LICENSE-MISMATCH, SSL-CARD-FAILED, SSL-CARD-NORMAL, WARM-RESTART-EVENT, HARD-DISK-DRIVE-ERRORS, COMPACT-FLASH-ERRORS, CALLHOME-UPLOAD-EVENT, 1024KEY-EXCHANGE-RATE, 2048KEY-EXCHANGE-RATE, 4096KEY-EXCHANGE-RATE, SSL-CUR-SESSION-INUSE, CLUSTER-NODE-HEALTH, CLUSTER-NODE-QUORUM, CLUSTER-VERSION-MISMATCH, CLUSTER-CCO-CHANGE, CLUSTER-OVS-CHANGE, CLUSTER-SYNC-FAILURE, CLUSTER-PROP-FAILURE, HA-STICKY-PRIMARY, INBAND-PROTOCOL-VERSION-MISMATCH, SSL-CHIP-REINIT, VRID-STATE-CHANGE, PORT-ALLOC-FAILED, LLDP-REMOTE-CHANGE, DUPLICATE-IPV6

Example

```
disable snmp alarm VSERVER-REQRATE disable snmp alarm CPU SYNFLOOD
```

show snmp alarm

Displays the settings of all SNMP alarms or of the specified SNMP alarm. To display the settings of all the SNMP alarms, run the command without any parameters. To display the settings of a particular SNMP alarm, specify the trapName (Alarm name) of the SNMP alarm.

Synopsys

```
show snmp alarm [<trapName>]
```

Arguments

trapName

Name of the SNMP alarm whose details you want the NetScaler appliance to display.

Possible values: CPU-USAGE, AVERAGE-CPU, MEMORY, MGMT-CPU-USAGE, SYNFLOOD, VSERVER-REQRATE, SERVICE-REQRATE, ENTITY-RXRATE, ENTITY-TXRATE, ENTITY-SYNFLOOD, SERVICE-MAXCLIENTS, HA-STATE-CHANGE, ENTITY-STATE, CONFIG-CHANGE, CONFIG-SAVE, SERVICEGROUP-MEMBER-REQRATE, SERVICEGROUP-MEMBER-MAXCLIENTS, MONITOR-RTO-THRESHOLD, LOGIN-FAILURE, SSL-CERT-EXPIRY, FAN-SPEED-LOW, VOLTAGE-LOW, VOLTAGE-HIGH, TEMPERATURE-HIGH, CPU-TEMPERATURE-HIGH, POWER-SUPPLY-FAILURE, DISK-USAGE-HIGH, INTERFACE-THROUGHPUT-LOW, MON_PROBE_FAILED, HA-VERSION-MISMATCH, HA-SYNC-FAILURE, HA-NO-HEARTBEATS, HA-BAD-SECONDARY-STATE, INTERFACE-BW-USAGE, RATE-LIMIT-THRESHOLD-EXCEEDED, ENTITY-NAME-CHANGE, HA-PROP-FAILURE, IP-CONFLICT, PF-RL-RATE-THRESHOLD, PF-RL-PPS-THRESHOLD, PF-RL-RATE-PKTS-DROPPED, PF-RL-PPS-PKTS-DROPPED, APPFW-START-URL, APPFW-DENY-URL, APPFW-VIOLATIONS-TYPE, APPFW-REFERER-HEADER, APPFW-CSRF-TAG, APPFW-COOKIE, APPFW-FIELD-CONSISTENCY, APPFW-BUFFER-OVERFLOW, APPFW-FIELD-FORMAT, APPFW-SAFE-COMMERCE, APPFW-SAFE-OBJECT, APPFW-POLICY-HIT, APPFW-XSS, APPFW-XML-XSS, APPFW-SQL, APPFW-XML-SQL, APPFW-XML-ATTACHMENT, APPFW-XML-DOS, APPFW-XML-VALIDATION, APPFW-XML-WSI, APPFW-XML-SCHEMA-COMPILE, APPFW-XML-SOAP-FAULT, DNSKEY-EXPIRY, HA-LICENSE-MISMATCH, SSL-CARD-FAILED, SSL-CARD-NORMAL, WARM-RESTART-EVENT, HARD-DISK-DRIVE-ERRORS, COMPACT-FLASH-ERRORS, CALLHOME-UPLOAD-EVENT, 1024KEY-EXCHANGE-RATE, 2048KEY-EXCHANGE-RATE, 4096KEY-EXCHANGE-RATE, SSL-CUR-SESSION-INUSE, CLUSTER-NODE-HEALTH, CLUSTER-NODE-QUORUM, CLUSTER-VERSION-MISMATCH, CLUSTER-CCO-CHANGE, CLUSTER-OVS-CHANGE, CLUSTER-SYNC-FAILURE, CLUSTER-PROP-FAILURE, HA-STICKY-PRIMARY, INBAND-PROTOCOL-VERSION-MISMATCH, SSL-CHIP-REINIT, VRID-STATE-CHANGE, PORT-ALLOC-FAILED, LLDP-REMOTE-CHANGE, DUPLICATE-IPV6

Outputs

thresholdValue

The high threshold value.

normalValue

The normal threshold value.

time

The time interval for the SYNFLOOD alarm.

state

Current state of the SNMP alarm. The NetScaler appliance generates trap messages only for SNMP alarms that are enabled. Some alarms are enabled by default, but you can disable them.

severity

The severity of this alarm.

logging

The log status of the alarm.

flags

timeout

If DB is enabled and clear config is fired, then to reset timeinterval of alarm, corresponding default time value is needed. This hidden argument holds the default time value for the corresponding alarm.

devno

count

stateflag

snmp community

The following operations can be performed on "snmp community":

[add](#) | [rm](#) | [show](#)

add snmp community

Creates an SNMP community, which is a password (string) used to authenticate SNMP queries from SNMP managers. You can associate it with any of the following SNMP query types: GET, GET NEXT, ALL, GET BULK. You can associate one or more community strings with each query type. For example, if you associate two community strings, such as Example and Test, with the query type GET NEXT, the NetScaler appliance considers only those GET NEXT SNMP query packets that contain Example or Test as the community string.

Synopsis

```
add snmp community <communityName> <permissions>
```

Arguments

communityName

The SNMP community string. Can consist of 1 to 31 characters that include uppercase and lowercase letters, numbers and special characters.

The following requirement applies only to the NetScaler CLI:

If the string includes one or more spaces, enclose the name in double or single quotation marks (for example, "my string" or 'my string').

permissions

The SNMP V1 or V2 query-type privilege that you want to associate with this SNMP community.

Possible values: GET, GET_NEXT, GET_BULK, SET, ALL

Example

```
add snmp community public ALL add snmp community a#12ab GET_BULK
```

rm snmp community

Removes an SNMP community from the NetScaler appliance. After you remove the SNMP community, the appliance does not respond to any SNMP queries that contain that community string.

Synopsis

```
rm snmp community <communityName>
```

Arguments

communityName

The name of the SNMP community.

Example

```
rm snmp community public
```

show snmp community

Displays the SNMP v1 or v2 query-type privileges (such as GET, GET NEXT, ALL, or GET BULK) that have been set for all SNMP communities or for the specified SNMP community. To display the settings of all the SNMP communities,

run the command without any parameters. To display the settings of a particular SNMP community, specify the name of the SNMP community.

Synopsys

`show snmp community [<communityName>]`

Arguments

communityName

The name of the SNMP community whose SNMP v1 or v2 query type privilege setting, such as GET, GET NEXT, ALL, or GET BULK, you want the NetScaler appliance to display.

Outputs

permissions

The SNMP V1 or V2 query-type privilege that you want to associate with this SNMP community.

devno

count

stateflag

Example

```
show snmp community
```

snmp engineId

The following operations can be performed on "snmp engineId":

[set](#) | [unset](#) | [show](#)

set snmp engineId

Modifies the SNMPv3 engine identification (ID) on the NetScaler appliance. Caution: Changing the ID of the SNMPv3 engine invalidates the current SNMP users. You have to reconfigure the SNMP users in the SNMP managers. The SNMPv3 engine has an identification (ID) that uniquely identifies it on the appliance and is used in the communication between the SNMPv3 user and the SNMPv3 engine. The engine ID is preconfigured by Citrix and is based on the MAC address of one of its interfaces. Overriding the engine ID is not necessary, but you can change it.

Synopsis

```
set snmp engineId <engineID> [-ownerNode <positive_integer>]
```

Arguments

engineID

A hexadecimal value of at least 10 characters, uniquely identifying the engineId

ownerNode

ID of the cluster node for which you are setting the engineid

Default value: -1

Minimum value: 0

Maximum value: 31

unset snmp engineId

Resets the SNMPv3 engine identification (ID) on the NetScaler appliance to its default value. The NetScaler appliance derives the engine ID from the MAC address of one of its interfaces. Caution: Changing the ID of the SNMPv3 engine invalidates the current SNMP users. You have to reconfigure the SNMP users in the SNMP managers..Refer to the set snmp engineId command for meanings of the arguments.

Synopsis

```
unset snmp engineId [-ownerNode <positive_integer>]
```

show snmp engineId

Displays the ID of the SNMPv3 engine of the NetScaler appliance.

Synopsis

```
show snmp engineId [-ownerNode <positive_integer>]
```

Arguments

ownerNode

ID of the cluster node for which you are setting the engineid

Default value: -1

Minimum value: 0

Maximum value: 31

Outputs

engineID

A hexadecimal value of at least 10 characters, uniquely identifying the engineid

defaultEngineID

Unique identifier to assign to the SNMPv3 engine. Should be a hexadecimal value with a minimum length of 10 hex characters.

devno

count

stateflag

snmp group

The following operations can be performed on "snmp group":

[add](#) | [rm](#) | [set](#) | [show](#)

add snmp group

Adds an SNMPv3 user group on the NetScaler appliance. SNMPv3 groups are logical aggregations of SNMPv3 users. SNMPv3 groups are used to implement access control and define the security levels for the users. You can add a maximum of 1000 SNMPv3 groups to the NetScaler appliance.

Synopsys

```
add snmp group <name> <securityLevel> -readViewName <string>
```

Arguments

name

Name for the SNMPv3 group. Can consist of 1 to 31 characters that include uppercase and lowercase letters, numbers, and the hyphen (-), period (.), pound (#), space (), at sign (@), equals (=), colon (:), and underscore (_) characters. You should choose a name that helps identify the SNMPv3 group.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose it in double or single quotation marks (for example, "my name" or 'my name').

securityLevel

Security level required for communication between the NetScaler appliance and the SNMPv3 users who belong to the group. Specify one of the following options:

noAuthNoPriv. Require neither authentication nor encryption.

authNoPriv. Require authentication but no encryption.

authPriv. Require authentication and encryption.

Note: If you specify authentication, you must specify an encryption algorithm when you assign an SNMPv3 user to the group. If you also specify encryption, you must assign both an authentication and an encryption algorithm for each group member.

Possible values: noAuthNoPriv, authNoPriv, authPriv

readViewName

Name of the configured SNMPv3 view that you want to bind to this SNMPv3 group. An SNMPv3 user bound to this group can access the subtrees that are bound to this SNMPv3 view as type INCLUDED, but cannot access the ones that are type EXCLUDED. If the NetScaler appliance has multiple SNMPv3 view entries with the same name, all such entries are associated with the SNMPv3 group.

rm snmp group

Removes an SNMPv3 group entry from the NetScaler appliance. The appliance can have multiple SNMPv3 groups with the same name, differentiated by the securityLevel (Security level) parameter setting. Therefore, to identify an SNMPv3 group entry that you want to remove, you have to specify both the name and security level of the SNMPv3 group.

Synopsys

```
rm snmp group <name> <securityLevel>
```

Arguments

name

Name of the SNMPv3 group.

securityLevel

Security level of the SNMPv3 group.

Possible values: noAuthNoPriv, authNoPriv, authPriv

set snmp group

Modifies the specified parameters of an SNMPv3 group entry on the NetScaler appliance.

Synopsis

```
set snmp group <name> <securityLevel> -readViewName <string>
```

Arguments

name

The name specified in the SNMPv3 group entry that you want to modify. This parameter cannot be modified.

securityLevel

Security level required for communication between the NetScaler appliance and the SNMPv3 users who belong to the group. Specify one of the following options:

noAuthNoPriv. Require neither authentication nor encryption.

authNoPriv. Require authentication but no encryption.

authPriv. Require authentication and encryption.

Note: If you specify authentication, you must specify an encryption algorithm when you assign an SNMPv3 user to the group. If you also specify encryption, you must assign both an authentication and an encryption algorithm for each group member.

Possible values: noAuthNoPriv, authNoPriv, authPriv

readViewName

Name of the configured SNMPv3 view that you want to bind to this SNMPv3 group. An SNMPv3 user bound to this group can access the subtrees that are bound to this SNMPv3 view as type INCLUDED, but cannot access the ones that are type EXCLUDED. If the NetScaler appliance has multiple SNMPv3 view entries with the same name, all such entries are associated with the SNMPv3 group.

show snmp group

Displays the settings of all SNMPv3 groups or of the specified SNMPv3 group. To display the settings of all SNMPv3 groups, run the command without any parameters. To display the settings of a particular SNMPv3 group, specify the name of the SNMPv3 group and securityLevel (Security level). The NetScaler appliance can have multiple SNMPv3 groups with the same name, differentiated by the securityLevel (Security level) parameter setting.

Synopsis

```
show snmp group [<name> <securityLevel>]
```

Arguments

name

Name of the SNMPv3 group whose details you want the NetScaler appliance to display.

securityLevel

Security level of the SNMPv3 group whose details you want the NetScaler appliance to display.

Possible values: noAuthNoPriv, authNoPriv, authPriv

Outputs

readViewName

Name of the configured SNMPv3 view that you want to bind to this SNMPv3 group. An SNMPv3 user bound to this group can access the subtrees that are bound to this SNMPv3 view as type INCLUDED, but cannot access the ones that are type EXCLUDED. If the NetScaler appliance has multiple SNMPv3 view entries with the same name, all such entries are associated with the SNMPv3 group.

storageType

The storage type for this group.

status

The status of this group.

devno

count

stateflag

snmp manager

The following operations can be performed on "snmp manager":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add snmp manager

Specifies an SNMP manager to query the NetScaler appliance. The added manager complies with SNMP V1, V2, and V3. If you specify one or more SNMP managers, the appliance does not accept SNMP queries from any hosts except the specified SNMP managers. You can specify up to a maximum of 100 IP based SNMP managers or networks and a maximum of 5 host-name based SNMP managers.

Synopsys

```
add snmp manager <IPAddress> ... [-netmask <netmask>] [-domainResolveRetry <integer>]
```

Arguments

IPAddress

IP address of the SNMP manager. Can be an IPv4 or IPv6 address. You can instead specify an IPv4 network address or IPv6 network prefix if you want the NetScaler appliance to respond to SNMP queries from any device on the specified network. Alternatively, instead of an IPv4 address, you can specify a host name that has been assigned to an SNMP manager. If you do so, you must add a DNS name server that resolves the host name of the SNMP manager to its IP address.

Note: The NetScaler appliance does not support host names for SNMP managers that have IPv6 addresses.

netmask

Subnet mask associated with an IPv4 network address. If the IP address specifies the address or host name of a specific host, accept the default value of 255.255.255.255.

Default value: 0xFFFFFFFF

domainResolveRetry

Amount of time, in seconds, for which the NetScaler appliance waits before sending another DNS query to resolve the host name of the SNMP manager if the last query failed. This parameter is valid for host-name based SNMP managers only. After a query succeeds, the TTL determines the wait time.

Minimum value: 5

Maximum value: 20939

Example

```
add snmp manager 192.168.1.20 192.168.2.42 add snmp manager 192.168.2.16 -netmask 255.255
```

rm snmp manager

Removes an SNMP manager from the list of managers that are allowed to access the NetScaler appliance.

Synopsys

```
rm snmp manager <IPAddress> ... [-netmask <netmask>]
```

Arguments

IPAddress

IPv4 or IPv6 address (or IPv4 host name) of the SNMP manager, or the IPv4 network address or IPv6 network prefix of the SNMP managers.

netmask

Subnet mask associated with an IPv4 SNMP manager entry. For a specific host, the subnet mask is 255.255.255.255.

Default value: 0xFFFFFFFF

Example

```
rm snmp manager 192.168.1.20 rm snmp manager 192.168.2.16 -netmask 255.255.255.240 rm snmp
```

set snmp manager

Modifies the Domain Resolve Retry parameter of any host-name based SNMP manager configured on the NetScaler appliance.

Synopsis

```
set snmp manager <IPAddress> [-netmask <netmask>] [-domainResolveRetry <integer>]
```

Arguments

IPAddress

Host name of the SNMP manager for which you want to modify the Domain Resolve Retry parameter.

netmask

Subnet mask associated with an IPv4 network address. If the IP address specifies the address or host name of a specific host, accept the default value of 255.255.255.255.

Default value: 0xFFFFFFFF

domainResolveRetry

Amount of time, in seconds, for which the NetScaler appliance waits before sending another DNS query to resolve the host name of the SNMP manager if the last query failed. This parameter is valid for host-name based SNMP managers only. After a query succeeds, the TTL determines the wait time.

Minimum value: 5

Maximum value: 20939

Example

```
set snmp manager www.example.com -domainResolveRetry 7
```

unset snmp manager

Use this command to remove snmp manager settings. Refer to the set snmp manager command for meanings of the arguments.

Synopsis

```
unset snmp manager <IPAddress> -netmask <netmask> -domainResolveRetry
```

show snmp manager

Displays configuration information about all SNMP managers on the NetScaler appliance, or detailed information about the specified manager.

Synopsis

```
show snmp manager [<IPAddress> [-netmask <netmask>]]
```

Arguments

IPAddress

IPv4 or IPv6 address (or IPv4 host name) of the SNMP manager, or the IPv4 network address or IPv6 network prefix of the SNMP managers, about which to display information.

netmask

Subnet mask associated with an IPv4-address based SNMP manager entry. If the IP address specifies a specific host (either the IP address or the host name of the SNMP manager), the subnet mask is 255.255.255.255.

Default value: 0

Outputs

IP

The resolved IP address of the hostname manager

domain

IP address of manager. It will be zero for hostname manager

domainResolveRetry

Amount of time, in seconds, for which the NetScaler appliance waits before sending another DNS query to resolve the host name of the SNMP manager if the last query failed. This parameter is valid for host-name based SNMP managers only. After a query succeeds, the TTL determines the wait time.

devno

count

stateflag

Example

```
show snmp manager
```

snmp mib

The following operations can be performed on "snmp mib":

[set](#) | [unset](#) | [show](#)

set snmp mib

Configures the SNMP agent of the NetScaler appliance with information that identifies the appliance, such as the name of the administrator for this NetScaler appliance, a name for the appliance, and the location of the appliance. SNMP managers can query the NetScaler appliance for this information.

Synopsys

```
set snmp mib [-contact <string>] [-name <string>] [-location <string>] [-customID <string>]
```

Arguments

contact

Name of the administrator for this NetScaler appliance. Along with the name, you can include information on how to contact this person, such as a phone number or an email address. Can consist of 1 to 127 characters that include uppercase and lowercase letters, numbers, and the hyphen (-), period (.), pound (#), space (), at sign (@), equals (=), colon (:), and underscore (_) characters.

The following requirement applies only to the NetScaler CLI:

If the information includes one or more spaces, enclose it in double or single quotation marks (for example, "my contact" or 'my contact').

Default value: "WebMaster (default)"

name

Name for this NetScaler appliance. Can consist of 1 to 127 characters that include uppercase and lowercase letters, numbers, and the hyphen (-), period (.), pound (#), space (), at sign (@), equals (=), colon (:), and underscore (_) characters. You should choose a name that helps identify the NetScaler appliance.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose it in double or single quotation marks (for example, "my name" or 'my name').

Default value: "NetScaler"

location

Physical location of the NetScaler appliance. For example, you can specify building name, lab number, and rack number. Can consist of 1 to 127 characters that include uppercase and lowercase letters, numbers, and the hyphen (-), period (.), pound (#), space (), at sign (@), equals (=), colon (:), and underscore (_) characters.

The following requirement applies only to the NetScaler CLI:

If the location includes one or more spaces, enclose it in double or single quotation marks (for example, "my location" or 'my location').

Default value: "POP (default)"

customID

Custom identification number for the NetScaler appliance. Can consist of 1 to 127 characters that include uppercase and lowercase letters, numbers, and the hyphen (-), period (.), pound (#), space (), at sign (@), equals (=), colon (:), and underscore (_) characters. You should choose a custom identification that helps identify the NetScaler appliance.

The following requirement applies only to the NetScaler CLI:

If the ID includes one or more spaces, enclose it in double or single quotation marks (for example, "my ID" or 'my ID').

Default value: "Default"

unset snmp mib

Use this command to remove snmp mib settings. Refer to the set snmp mib command for meanings of the arguments.

Synopsys

```
unset snmp mib [-contact] [-name] [-location] [-customID]
```

show snmp mib

Displays the information that has been configured on the SNMP agent for the purpose of identifying the NetScaler appliance, such as the name of the appliance, administrator, and location.

Synopsys

```
show snmp mib
```

Outputs

contact

Name of the administrator for this NetScaler appliance. Along with the name, you can include information on how to contact this person, such as a phone number or an email address. Can consist of 1 to 127 characters that include uppercase and lowercase letters, numbers, and the hyphen (-), period (.), pound (#), space (), at sign (@), equals (=), colon (:), and underscore (_) characters.

The following requirement applies only to the NetScaler CLI:

If the information includes one or more spaces, enclose it in double or single quotation marks (for example, "my contact" or 'my contact').

name

Name for this NetScaler appliance. Can consist of 1 to 127 characters that include uppercase and lowercase letters, numbers, and the hyphen (-), period (.), pound (#), space (), at sign (@), equals (=), colon (:), and underscore (_) characters. You should choose a name that helps identify the NetScaler appliance.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose it in double or single quotation marks (for example, "my name" or 'my name').

location

Physical location of the NetScaler appliance. For example, you can specify building name, lab number, and rack number. Can consist of 1 to 127 characters that include uppercase and lowercase letters, numbers, and the hyphen (-), period (.), pound (#), space (), at sign (@), equals (=), colon (:), and underscore (_) characters.

The following requirement applies only to the NetScaler CLI:

If the location includes one or more spaces, enclose it in double or single quotation marks (for example, "my location" or 'my location').

sysDesc

The description of the system.

sysUptime

The UP time of the system in 100th of a second.

sysServices

The services offered by the system.

sysOID

The OID of the system's management system.

customID

Custom identification number for the NetScaler appliance. Can consist of 1 to 127 characters that include uppercase and lowercase letters, numbers, and the hyphen (-), period (.), pound (#), space (), at sign (@), equals (=), colon (:), and underscore (_) characters. You should choose a custom identification that helps identify the NetScaler appliance.

The following requirement applies only to the NetScaler CLI:

If the ID includes one or more spaces, enclose it in double or single quotation marks (for example, "my ID" or 'my ID').

Example

```
show snmp mib
```


snmp oid

The following operations can be performed on "snmp oid":

show snmp oid

Displays the corresponding SNMP OIDs for the virtual servers, services, and service groups configured on the NetScaler appliance. To display the SNMP OID of all entities of a particular type, such as virtual servers, run the command with only that entity type specified. To display the SNMP of a particular entity, specify the entity type and the entity name.

Synopsys

```
show snmp oid <entityType> [<name>]
```

Arguments

entityType

The type of entity whose SNMP OIDs you want to displayType of entity whose SNMP OIDs you want the NetScaler appliance to display.

Possible values: VSERVER, SERVICE, SERVICEGROUP

name

Name of the entity whose SNMP OID you want the NetScaler appliance to display.

Outputs

snmpOID

The snmp oid.

stateflag

state flag

devno

count

Example

```
show snmp oid VSERVER vs1
```

snmp option

The following operations can be performed on "snmp option":

[set](#) | [unset](#) | [show](#)

set snmp option

Enables or disables SNMP options for SNMP SET and SNMP trap logging.

Synopsys

```
set snmp option [-snmpset ( ENABLED | DISABLED )] [-snmpTrapLogging ( ENABLED | DISABLED )]
```

Arguments

snmpset

Accept SNMP SET requests sent to the NetScaler appliance, and allow SNMP managers to write values to MIB objects that are configured for write access.

Possible values: ENABLED, DISABLED

Default value: DISABLED

snmpTrapLogging

Log any SNMP trap events (for SNMP alarms in which logging is enabled) even if no trap listeners are configured. With the default setting, SNMP trap events are logged if at least one trap listener is configured on the appliance.

Possible values: ENABLED, DISABLED

Default value: DISABLED

unset snmp option

Use this command to remove snmp option settings. Refer to the set snmp option command for meanings of the arguments.

Synopsys

```
unset snmp option [-snmpset] [-snmpTrapLogging]
```

show snmp option

Displays the settings for the following SNMP options: SNMP SET and SNMP trap Logging.

Synopsys

```
show snmp option
```

Outputs

snmpset

Accept SNMP SET requests sent to the NetScaler appliance, and allow SNMP managers to write values to MIB objects that are configured for write access.

snmpTrapLogging

Log any SNMP trap events (for SNMP alarms in which logging is enabled) even if no trap listeners are configured. With the default setting, SNMP trap events are logged if at least one trap listener is configured on the appliance.

snmp stats

The following operations can be performed on "snmp stats":

show snmp stats

show snmp stats is an alias for stat snmp Displays the statistics related to SNMP.

Synopsys

show snmp stats - alias for 'stat snmp'

snmp trap

The following operations can be performed on "snmp trap":

add | **rm** | **set** | **unset** | **show** | **bind** | **unbind**

add snmp trap

Adds an SNMP trap listener. You can configure the NetScaler appliance to generate asynchronous events (trap messages) to report abnormal conditions. The trap messages are sent to a remote device (trap listener) to help administrators monitor the appliance and respond promptly to any issues.

Synopsys

```
add snmp trap <trapClass> <trapDestination> ... [-version <version>] [-td <positive_integer>] [-destPort <port>] [-communityName <string>] [-srcIP <ip_addr|ipv6_addr>] [-severity <severity>]
```

Arguments

trapClass

Type of trap messages that the NetScaler appliance sends to the trap listener: Generic or the enterprise-specific messages defined in the MIB file.

Possible values: generic, specific

trapDestination

IPv4 or the IPv6 address of the trap listener to which the NetScaler appliance is to send SNMP trap messages.

version

SNMP version, which determines the format of trap messages sent to the trap listener.

This setting must match the setting on the trap listener. Otherwise, the listener drops the trap messages.

Possible values: V1, V2, V3

Default value: V2

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

destPort

UDP port at which the trap listener listens for trap messages. This setting must match the setting on the trap listener. Otherwise, the listener drops the trap messages.

Default value: 162

Minimum value: 1

Maximum value: 65534

communityName

Password (string) sent with the trap messages, so that the trap listener can authenticate them. Can include 1 to 31 uppercase or lowercase letters, numbers, and hyphen (-), period (.) pound (#), space (), at (@), equals (=), colon (:), and underscore (_) characters.

You must specify the same community string on the trap listener device. Otherwise, the trap listener drops the trap messages.

The following requirement applies only to the NetScaler CLI:

If the string includes one or more spaces, enclose the name in double or single quotation marks (for example, "my string" or 'my string').

srcIP

IPv4 or IPv6 address that the NetScaler appliance inserts as the source IP address in all SNMP trap messages that it sends to this trap listener. By default this is the appliance's NSIP or NSIP6 address, but you can specify an IPv4 MIP or SNIP address or a SNIP6 address.

severity

Severity level at or above which the NetScaler appliance sends trap messages to this trap listener. The severity levels, in increasing order of severity, are Informational, Warning, Minor, Major, Critical. This parameter can be set for trap listeners of type SPECIFIC only. The default is to send all levels of trap messages.

Important: Trap messages are not assigned severity levels unless you specify severity levels when configuring SNMP alarms.

Possible values: Critical, Major, Minor, Warning, Informational

Default value: Unknown

rm snmp trap

Removes a trap listener entry from the NetScaler appliance.

Synopsis

```
rm snmp trap <trapClass> <trapDestination> ... [-version <version>] [-td <positive_integer>]
```

Arguments

trapClass

Trap type specified in the trap listener entry that you want to remove.

Possible values: generic, specific

trapDestination

IP address of the trap listener specified in the trap listener entry that you want to remove.

version

Version of the trap specified in the trap listener entry that you want to remove.

Possible values: V1, V2, V3

Default value: V2

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

set snmp trap

Modifies the specified parameters in a trap-listener entry.

Synopsys

```
set snmp trap <trapClass> <trapDestination> [-version <version>] [-td <positive_integer>] [-destPort <port>] [-communityName <string>] [-srcIP <ip_addr|ipv6_addr>] [-severity <severity>]
```

Arguments

trapClass

Type of trap specified in the trap-listener entry. Because this parameter is used for identifying the trap listener entry, it cannot be modified after the entry has been created.

Possible values: generic, specific

trapDestination

IPv4 or the IPv6 address of the trap listener to which the NetScaler appliance is to send SNMP trap messages.

version

SNMP version, which determines the format of trap messages sent to the trap listener.

This setting must match the setting on the trap listener. Otherwise, the listener drops the trap messages.

Possible values: V1, V2, V3

Default value: V2

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

destPort

UDP port at which the trap listener listens for trap messages. This setting must match the setting on the trap listener. Otherwise, the listener drops the trap messages.

Default value: 162

Minimum value: 1

Maximum value: 65534

communityName

Password (string) sent with the trap messages, so that the trap listener can authenticate them. Can include 1 to 31 uppercase or lowercase letters, numbers, and hyphen (-), period (.) pound (#), space (), at (@), equals (=), colon (:), and underscore (_) characters.

You must specify the same community string on the trap listener device. Otherwise, the trap listener drops the trap messages.

The following requirement applies only to the NetScaler CLI:

If the string includes one or more spaces, enclose the name in double or single quotation marks (for example, "my string" or 'my string').

srcIP

IPv4 or IPv6 address that the NetScaler appliance inserts as the source IP address in all SNMP trap messages that it sends to this trap listener. By default this is the appliance's NSIP or NSIP6 address, but you can specify an IPv4 MIP or SNIP address or a SNIP6 address.

severity

Severity level at or above which the NetScaler appliance sends trap messages to this trap listener. The severity levels, in increasing order of severity, are Informational, Warning, Minor, Major, Critical. This parameter can be set for trap listeners of type SPECIFIC only. The default is to send all levels of trap messages.

Important: Trap messages are not assigned severity levels unless you specify severity levels when configuring SNMP alarms.

Possible values: Critical, Major, Minor, Warning, Informational

Default value: Unknown

Example

```
set snmp trap generic 192.168.3.4 -version V1 -severity Critical
```

unset snmp trap

Resets the specified parameters to their default settings in a trap-listener entry..Refer to the set snmp trap command for meanings of the arguments.

Synopsys

```
unset snmp trap <trapClass> <trapDestination> [-version <version>] [-td <positive_integer>] [-destPort] [-communityName] [-srcIP] [-severity]
```

Example

```
unset snmp trap generic 192.168.3.4 -version V1 -severity
```

show snmp trap

Displays the settings of all trap listeners or of the specified trap listener. To display the settings of all the trap listeners, run the command without any parameters. To display the settings of a particular trap listener, specify the trapClass (Trap Type) and trapDestination (IP Address) of the trap listener.

Synopsys

```
show snmp trap [<trapClass> <trapDestination> [-version <version>] [-td <positive_integer>]]
```

Arguments

trapClass

Trap type specified in the trap listener entry.

Possible values: generic, specific

trapDestination

IP address specified in the trap listener entry.

version

The SNMP version of the trap specified in the trap listener entry.

Possible values: V1, V2, V3

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

Outputs

destPort

The destination port of the SNMP trap.

communityName

Password (string) sent with the trap messages, so that the trap listener can authenticate them. Can include 1 to 31 uppercase or lowercase letters, numbers, and hyphen (-), period (.) pound (#), space (), at (@), equals (=), colon (:), and underscore (_) characters.

You must specify the same community string on the trap listener device. Otherwise, the trap listener drops the trap messages.

The following requirement applies only to the NetScaler CLI:

If the string includes one or more spaces, enclose the name in double or single quotation marks (for example, "my string" or 'my string').

srcIP

The source IP of the SNMP trap to be sent.

severity

The minimum severity of traps to be sent to this destination.

userName

Name of the SNMP user that will send the SNMPv3 traps.

securityLevel

Security level of the SNMPv3 trap.

stateflag

devno

count

Example

```
show snmp trap
```

bind snmp trap

Binds an SNMPv3 trap to an SNMP user.

Synopsys

```
bind snmp trap <trapClass> <trapDestination> [-td <positive_integer>] [-version <version>] [-userName <string> [-securityLevel <securityLevel>]]
```

Arguments

trapClass

Type of trap messages that the NetScaler appliance sends to the trap listener: Generic or the enterprise-specific messages defined in the MIB file.

Possible values: generic, specific

trapDestination

IPv4 or the IPv6 address of the trap listener to which the NetScaler appliance is to send SNMP trap messages.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

version

SNMP version, which determines the format of trap messages sent to the trap listener.

This setting must match the setting on the trap listener. Otherwise, the listener drops the trap messages.

Possible values: V1, V2, V3

Default value: V3

userName

Name of the SNMP user that will send the SNMPv3 traps.

securityLevel

Security level of the SNMPv3 trap.

Possible values: noAuthNoPriv, authNoPriv, authPriv

Default value: authNoPriv,

unbind snmp trap

Unbind snmp user to a V3 trap

Synopsys

```
unbind snmp trap <trapClass> <trapDestination> [-td <positive_integer>] [-version <version>] -userName <string>
```

Arguments

trapClass

Type of trap messages that the NetScaler appliance sends to the trap listener: Generic or the enterprise-specific messages defined in the MIB file.

Possible values: generic, specific

trapDestination

IPv4 or the IPv6 address of the trap listener to which the NetScaler appliance is to send SNMP trap messages.

td

Integer value that uniquely identifies the traffic domain in which you want to configure the entity. If you do not specify an ID, the entity becomes part of the default traffic domain, which has an ID of 0.

Minimum value: 0

Maximum value: 4094

version

SNMP version, which determines the format of trap messages sent to the trap listener.

This setting must match the setting on the trap listener. Otherwise, the listener drops the trap messages.

Possible values: V1, V2, V3

Default value: V3

userName

Name of the SNMP user that will send the SNMPv3 traps.

snmp user

The following operations can be performed on "snmp user":

add | rm | set | unset | show

add snmp user

Adds an SNMPv3 user who can send SNMP queries to the NetScaler appliance. You can add a maximum of 1000 SNMPv3 users.

Synopsys

```
add snmp user <name> -group <string> [-authType ( MD5 | SHA ) {-authPasswd } [-privType ( DES | AES ) {-privPasswd }]]
```

Arguments

name

Name for the SNMPv3 user. Can consist of 1 to 31 characters that include uppercase and lowercase letters, numbers, and the hyphen (-), period (.) pound (#), space (), at sign (@), equals (=), colon (:), and underscore (_) characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose it in double or single quotation marks (for example, "my user" or 'my user').

group

Name of the configured SNMPv3 group to which to bind this SNMPv3 user. The access rights (bound SNMPv3 views) and security level set for this group are assigned to this user.

authType

Authentication algorithm used by the NetScaler appliance and the SNMPv3 user for authenticating the communication between them. You must specify the same authentication algorithm when you configure the SNMPv3 user in the SNMP manager.

Possible values: MD5, SHA

authPasswd

Plain-text pass phrase to be used by the authentication algorithm specified by the authType (Authentication Type) parameter. Can consist of 1 to 31 characters that include uppercase and lowercase letters, numbers, and the hyphen (-), period (.) pound (#), space (), at sign (@), equals (=), colon (:), and underscore (_) characters.

The following requirement applies only to the NetScaler CLI:

If the pass phrase includes one or more spaces, enclose it in double or single quotation marks (for example, "my phrase" or 'my phrase').

privType

Encryption algorithm used by the NetScaler appliance and the SNMPv3 user for encrypting the communication between them. You must specify the same encryption algorithm when you configure the SNMPv3 user in the SNMP manager.

Possible values: DES, AES

privPasswd

Encryption key to be used by the encryption algorithm specified by the privType (Encryption Type) parameter. Can consist of 1 to 31 characters that include uppercase and lowercase letters, numbers, and the hyphen (-), period (.) pound (#), space (), at sign (@), equals (=), colon (:), and underscore (_) characters.

The following requirement applies only to the NetScaler CLI:

If the key includes one or more spaces, enclose it in double or single quotation marks (for example, "my key" or 'my key').

rm snmp user

Removes an SNMPv3 user entry from the NetScaler appliance.

Synopsis

```
rm snmp user <name>
```

Arguments

name

Name of the SNMPv3 user.

set snmp user

Modifies the specified parameters of an SNMPv3 user entry on the NetScaler appliance.

Synopsis

```
set snmp user <name> [-group <string>] [-authType ( MD5 | SHA ) {-authPasswd }] [-privType ( DES | AES ) {-privPasswd }]
```

Arguments

name

Name specified in the SNMPv3 user entry that you want to modify. Because this parameter is used for identifying the SNMPv3 user entry, it cannot be modified after the entry has been created.

group

Name of the configured SNMPv3 group to which to bind this SNMPv3 user. The access rights (bound SNMPv3 views) and security level set for this group are assigned to this user.

authType

Authentication algorithm used by the NetScaler appliance and the SNMPv3 user for authenticating the communication between them. You must specify the same authentication algorithm when you configure the SNMPv3 user in the SNMP manager.

Possible values: MD5, SHA

authPasswd

Plain-text pass phrase to be used by the authentication algorithm specified by the authType (Authentication Type) parameter. Can consist of 1 to 31 characters that include uppercase and lowercase letters, numbers, and the hyphen (-), period (.) pound (#), space (), at sign (@), equals (=), colon (:), and underscore (_) characters.

The following requirement applies only to the NetScaler CLI:

If the pass phrase includes one or more spaces, enclose it in double or single quotation marks (for example, "my phrase" or 'my phrase').

privType

Encryption algorithm used by the NetScaler appliance and the SNMPv3 user for encrypting the communication between them. You must specify the same encryption algorithm when you configure the SNMPv3 user in the SNMP manager.

Possible values: DES, AES

privPasswd

Encryption key to be used by the encryption algorithm specified by the privType (Encryption Type) parameter. Can consist of 1 to 31 characters that include uppercase and lowercase letters, numbers, and the hyphen (-), period (.), pound (#), space (), at sign (@), equals (=), colon (:), and underscore (_) characters.

The following requirement applies only to the NetScaler CLI:

If the key includes one or more spaces, enclose it in double or single quotation marks (for example, "my key" or 'my key').

unset snmp user

Resets the specified parameters of an SNMPv3 user entry to their default settings..Refer to the set snmp user command for meanings of the arguments.

Synopsys

```
unset snmp user <name> [-authType | -privType] [-authPasswd] [-privPasswd]
```

show snmp user

Displays the settings of all SNMPv3 users or of the specified SNMPv3 user. To display the settings of all the SNMPv3 users, run the command without any parameters. To display the settings of a particular SNMPv3 user, specify the name of the SNMPv3 user.

Synopsys

```
show snmp user [<name>]
```

Arguments

name

Name of the SNMPv3 user whose details you want the NetScaler appliance to display.

Outputs

group

Name of the configured SNMPv3 group to which to bind this SNMPv3 user. The access rights (bound SNMPv3 views) and security level set for this group are assigned to this user.

authType

Authentication algorithm used by the NetScaler appliance and the SNMPv3 user for authenticating the communication between them. You must specify the same authentication algorithm when you configure the SNMPv3 user in the SNMP manager.

authPasswd

Plain-text pass phrase to be used by the authentication algorithm specified by the authType (Authentication Type) parameter. Can consist of 1 to 31 characters that include uppercase and lowercase letters, numbers, and the hyphen (-), period (.) pound (#), space (), at sign (@), equals (=), colon (:), and underscore (_) characters.

The following requirement applies only to the NetScaler CLI:

If the pass phrase includes one or more spaces, enclose it in double or single quotation marks (for example, "my phrase" or 'my phrase').

privType

Encryption algorithm used by the NetScaler appliance and the SNMPv3 user for encrypting the communication between them. You must specify the same encryption algorithm when you configure the SNMPv3 user in the SNMP manager.

privPasswd

Encryption key to be used by the encryption algorithm specified by the privType (Encryption Type) parameter. Can consist of 1 to 31 characters that include uppercase and lowercase letters, numbers, and the hyphen (-), period (.) pound (#), space (), at sign (@), equals (=), colon (:), and underscore (_) characters.

The following requirement applies only to the NetScaler CLI:

If the key includes one or more spaces, enclose it in double or single quotation marks (for example, "my key" or 'my key').

engineID

The context engine ID of the user.

storageType

The storage type for this user.

status

The status of this user.

devno**count****stateflag**

snmp view

The following operations can be performed on "snmp view":

[add](#) | [rm](#) | [set](#) | [show](#)

add snmp view

Adds an SNMPv3 view. Used to implement access control for the SNMPv3 user, SNMPv3 views restrict user access to specific portions of the MIB. The NetScaler appliance can have multiple SNMPv3 views with the same name, differentiated by subtree parameter settings. You can add a maximum of 1000 SNMPv3 views.

Synopsys

```
add snmp view <name> <subtree> -type ( included | excluded )
```

Arguments

name

Name for the SNMPv3 view. Can consist of 1 to 31 characters that include uppercase and lowercase letters, numbers, and the hyphen (-), period (.) pound (#), space (), at sign (@), equals (=), colon (:), and underscore (_) characters. You should choose a name that helps identify the SNMPv3 view.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose it in double or single quotation marks (for example, "my view" or 'my view').

subtree

A particular branch (subtree) of the MIB tree that you want to associate with this SNMPv3 view. You must specify the subtree as an SNMP OID.

type

Include or exclude the subtree, specified by the subtree parameter, in or from this view. This setting can be useful when you have included a subtree, such as A, in an SNMPv3 view and you want to exclude a specific subtree of A, such as B, from the SNMPv3 view.

Possible values: included, excluded

rm snmp view

Removes an SNMPv3 view entry from the NetScaler appliance. The appliance can have multiple SNMPv3 views with the same name, differentiated by the subtree parameter setting. Therefore, to identify an SNMPv3 group subtree that you want to remove, you have to specify both the name and subtree of the SNMPv3 view.

Synopsys

```
rm snmp view <name> <subtree>
```

Arguments

name

Name of the SNMPv3 view. Note: If multiple views have the same name, specify the subtree to identify the view to be removed.

subtree

A MIB subtree of the SNMPv3 view.

set snmp view

Modifies the type (Type) parameter of an SNMPv3 view configured on the NetScaler appliance.

Synopsys

```
set snmp view <name> <subtree> -type ( included | excluded )
```

Arguments

name

The name specified in the SNMPv3 view entry. This parameter cannot be modified.

subtree

A MIB subtree of the SNMPv3 view entry. This parameter cannot be modified.

type

Include or exclude the subtree, specified by the subtree parameter, in or from this view. This setting can be useful when you have included a subtree, such as A, in an SNMPv3 view and you want to exclude a specific subtree of A, such as B, from the SNMPv3 view.

Possible values: included, excluded

show snmp view

Displays the settings of all SNMPv3 views or of the specified SNMPv3 view. To display the settings of all the SNMPv3 views, run the command without any parameters. To display the settings of a particular SNMPv3 view, specify the name of the SNMPv3 view and subtree (the associated subtree of the MIB). The NetScaler appliance can have multiple SNMPv3 views with the same name, differentiated by the subtree parameter settings.

Synopsys

```
show snmp view [<name> [<subtree>]]
```

Arguments

name

Name of the SNMPv3 view.

subtree

A MIB subtree of the SNMPv3 view.

Outputs

type

The type of subtree.

storageType

The storage type for this view.

status

The status of this view.

devno

count

stateflag

Spillover Commands

The entities on which you can perform NetScaler CLI operations:

- [spillover action](#)
- [spillover policy](#)

spillover action

The following operations can be performed on "spillover action":

[add](#) | [rm](#) | [show](#) | [rename](#)

add spillover action

Creating spillover action

Synopsys

add spillover action <name> -action SPILLOVER

Arguments

name

Name of the spillover action.

action

Spillover action. Currently only type SPILLOVER is supported

Possible values: SPILLOVER

rm spillover action

Removes a spillover policy.

Synopsys

rm spillover action <name>

Arguments

name

Name of the spillover action.

show spillover action

Displaying spillover actions

Synopsys

show spillover action [<name>]

Arguments

name

Name of the spillover action.

Outputs

action

Spillover action. Currently only type SPILLOVER is supported

builtin

Flag to determine whether compression is default or not

gslbBindings

Number of times action is used in a policy bound to a gslb vserver

devno

count

stateflag

rename spillover action

Renames a spillover action.

Synopsis

```
rename spillover action <name>@ <newName>@
```

Arguments

name

Existing name of the action.

newName

New name for the spillover action. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at

(@), equals (=), and hyphen (-) characters.

Choose a name that can be correlated with the function that the action performs.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my action" or 'my action').

Example

```
rename spillover policy oldname newname
```

spillover policy

The following operations can be performed on "spillover policy":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#) | [rename](#) | [stat](#)

add spillover policy

Add a spillover policy. SPILLOVER policies that can be added are based on vserver expressions.

Synopsys

```
add spillover policy <name> -rule <expression> -action <string> [-comment <string>]
```

Arguments

name

Name of the spillover policy.

rule

Expression to be used by the spillover policy.

action

Action for the spillover policy. Action is created using add spillover action command

comment

Any comments that you might want to associate with the spillover policy.

Example

```
add spillover policy poll -rule "SYS.VSERVER("abc").ACTIVESERVICES.LE(2) -action act1 add
```

rm spillover policy

Removes a spillover policy.

Synopsys

```
rm spillover policy <name>
```

Arguments

name

Name of the spillover policy.

set spillover policy

Used to change the expression or other parameters of an existing policy.

Synopsys

```
set spillover policy <name> [-rule <expression>] [-action <string>] [-comment <string>]
```

Arguments

name

Name of the spillover policy.

rule

Expression to be used by the spillover policy.

action

Action for the spillover policy. Action is created using add spillover action command

comment

Any comments that you might want to associate with the spillover policy.

Example

```
set spillover policy poll -rule "SYS.VSERVER("abc").ACTIVSERVICS.LE(1)" set spillover po:
```

unset spillover policy

Use this command to remove spillover policy settings. Refer to the set spillover policy command for meanings of the arguments.

Synopsys

```
unset spillover policy <name> -comment
```

show spillover policy

Displaying the policy-related information.

Synopsys

```
show spillover policy [<name>]
```

Arguments

name

Name of the spillover policy.

Outputs

stateflag**rule**

Expression to be used by the spillover policy.

action

Action for the spillover policy. Action is created using add spillover action command

boundTo

The name of the entity to which the policy is bound.

hits

The number of times the policy has been hit.

undefHits

Number of policy UNDEF hits.

activePolicy

Indicates whether policy is bound or not.

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

labelType

Type of policy label invocation.

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

comment

Any comments that you might want to associate with the spillover policy.

builtin

Flag to determine if compression policy is builtin or not

gslbBindings

Number of times the policy bound to a gslb vserver

devno**count**

rename spillover policy

Renames a spillover policy.

Synopsis

```
rename spillover policy <name>@ <newName>@
```

Arguments

name

Existing name of the policy.

newName

New name for the spillover policy. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

Choose a name that reflects the function that the policy performs.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my policy" or 'my policy').

Example

```
rename spillover policy oldname newname
```

stat spillover policy

Displays statistics for all spillover policies currently configured on the NetScaler appliance, or detailed statistics for the specified policy.

Synopsys

```
stat spillover policy [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

name

Name of the spillover policy for which to show detailed statistics.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Policy hits (Hits)

Number of hits on the policy

Policy undef hits (Undefhits)

Number of undef hits on the policy

SSL Commands

The entities on which you can perform NetScaler CLI operations:

- o ssl
- o ssl action
- o ssl cert
- o ssl certChain
- o ssl certFile
- o ssl certKey
- o ssl certLink
- o ssl certReq
- o ssl cipher
- o ssl ciphersuite
- o ssl crl
- o ssl crlFile
- o ssl dhFile
- o ssl dhParam
- o ssl dsaKey
- o ssl dtlsProfile
- o ssl fips
- o ssl fipsKey
- o ssl fipsSIMSource
- o ssl fipsSIMTarget
- o ssl global
- o ssl keyFile
- o ssl ocsponder
- o ssl parameter
- o ssl pkcs12
- o ssl pkcs8
- o ssl policy
- o ssl policylabel
- o ssl profile
- o ssl rsaKey
- o ssl service
- o ssl serviceGroup
- o ssl stats
- o ssl vserver
- o ssl wrapkey

ssl

The following operations can be performed on "ssl":

stat ssl

Displays SSL statistics.

Synopsys

```
stat ssl [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

SSL cards UP (SSLCardUP)

Number of SSL cards that are UP. If the number of cards UP is lower than a threshold, a failover is initiated.

SSL crypto card status (SSLCardSt)

Status of the SSL card(s). The value should be interpreted in binary form, with each set bit indicates a card as UP.

SSL cards present (SSLCards)

Number of SSL crypto cards present on the NetScaler appliance.

SSL engine status (SSLEngSt)

State of the SSL Engine (1=UP/0=DOWN). This state is decided based on SSL Feature/License status and minimum number of cards UP.

SSL sessions (SSLSe)

Number of SSL sessions on the NetScaler appliance.

SSL transactions (SSLTrn)

Number of SSL transactions on the NetScaler appliance.

SSLv2 transactions (SSL2Trn)

Number of SSLv2 transactions on the NetScaler appliance.

SSLv3 transactions (SSL3Trn)

Total number of SSLv3 transactions on the NetScaler appliance.

TLSv1 transactions (TLS1Trn)

Number of TLSv1 transactions on the NetScaler appliance.

TLSv1.1 transactions (TLS11Trn)

Number of TLSv1.1 transactions on the NetScaler appliance.

TLSv1.2 transactions (TLS12Trn)

Number of TLSv1.2 transactions on the NetScaler appliance.

SSLv2 sessions (SSL2Se)

Number of SSLv2 sessions on the NetScaler appliance.

SSLv3 sessions (SSL3Se)

Number of SSLv3 sessions on the NetScaler appliance.

TLSv1 sessions (TLS1Se)

Number of TLSv1 sessions on the NetScaler appliance.

TLSv1.1 sessions (TLS11Se)

Number of TLSv1.1 sessions on the NetScaler appliance.

TLSv1.2 sessions (TLS12Se)

Number of TLSv1.2 sessions on the NetScaler appliance.

new SSL sessions (NewSe)

Number of new SSL sessions created on the NetScaler appliance.

SSL session misses (SeMiss)

Number of SSL session reuse misses on the NetScaler appliance.

SSL session hits (SeHit)

Number of SSL session reuse hits on the NetScaler appliance.

SSL sessions (BSSLSe)

Number of back-end SSL sessions on the NetScaler appliance.

SSLv3 sessions (BSSL3Se)

Number of back-end SSLv3 sessions on the NetScaler appliance.

TLSv1 sessions (BTLS1Se)

Number of back-end TLSv1 sessions on the NetScaler appliance.

TLSv1.1 sessions (BTLS1Se)

Number of back-end TLSv1.1 sessions on the NetScaler appliance.

TLSv1.2 sessions (BTLS1Se)

Number of back-end TLSv1.2 sessions on the NetScaler appliance.

Session multiplex attempts (BSeMx)

Number of back-end SSL session multiplex attempts on the NetScaler appliance.

Session multiplex successes (BSeMxS)

Number of back-end SSL session multiplex successes on the NetScaler appliance.

Session multiplex failures (BSeMxF)

Number of back-end SSL session multiplex failures on the NetScaler appliance.

Bytes encrypted (Enc)

Number of bytes encrypted on the NetScaler appliance.

Bytes decrypted (Dec)

Number of bytes decrypted on the NetScaler appliance.

SSL session renegotiations (SSLRn)

Number of SSL session renegotiations on the NetScaler appliance.

SSLv3 session renegotiations (SSL3Rn)

Number of session renegotiations done on SSLv3.

TLSv1 session renegotiations (TLS1Rn)

Number of SSL session renegotiations done on TLSv1.

TLSv1.1 session renegotiations (TLS11Rn)

Number of SSL session renegotiations done on TLSv1.1.

TLSv1.2 session renegotiations (TLS12Rn)

Number of SSL session renegotiations done on TLSv1.2.

RSA 512-bit key exchanges (RSAKx5)

Number of RSA 512-bit key exchanges on the NetScaler appliance.

RSA 1024-bit key exchanges (RSAKx1)

Number of RSA 1024-bit key exchanges on the NetScaler appliance.

RSA 2048-bit key exchanges (RSAKx2)

Number of RSA 2048-bit key exchanges on the NetScaler appliance.

RSA 4096-bit key exchanges (RSAKx4)

Number of RSA 4096-bit key exchanges on the NetScaler appliance.

DH 512-bit key exchanges (DHKx5)

Number of Diffie-Helman 512-bit key exchanges on the NetScaler appliance.

DH 1024-bit key exchanges (DHKx1)

Number of Diffie-Helman 1024-bit key exchanges on the NetScaler appliance.

DH 2048-bit key exchanges (DHKx2)

Number of Diffie-Helman 2048-bit key exchanges on the NetScaler appliance.

ECDHE 521 curve key exchanges (ECDHEKx521)

Number of 521 Elliptical Curve Diffie-Helman on the NetScaler appliance.

ECDHE 384 curve key exchanges (ECDHEKx384)

Number of 384 Elliptical Curve Diffie-Helman on the NetScaler appliance.

ECDHE 256 curve key exchanges (ECDHEKx256)

Number of 256 Elliptical Curve Diffie-Helman on the NetScaler appliance.

ECDHE 224 curve key exchanges (ECDHEKx224)

Number of 224 Elliptical Curve Diffie-Helman on the NetScaler appliance.

RC4 40-bit encryptions (RC4En4)

Number of RC4 40-bit cipher encryptions on the NetScaler appliance.

RC4 56-bit encryptions (RC4En5)

Number of RC4 56-bit cipher encryptions on the NetScaler appliance.

RC4 64-bit encryptions (RC4En6)

Number of RC4 64-bit cipher encryptions on the NetScaler appliance.

RC4 128-bit encryptions (RC4En1)

Number of RC4 128-bit cipher encryptions on the NetScaler appliance.

DES 40-bit encryptions (DESEn4)

Number of DES 40-bit cipher encryptions on the NetScaler appliance.

DES 56-bit encryptions (DESEn5)

Number of DES 56-bit cipher encryptions on the NetScaler appliance.

3DES 168-bit encryptions (3DESEn1)

Number of DES 168-bit cipher encryptions on the NetScaler appliance.

AES 128-bit encryptions (AESEn1)

Number of AES 128-bit cipher encryptions on the NetScaler appliance.

AES 256-bit encryptions (AESEn2)

Number of AES 256-bit cipher encryptions on the NetScaler appliance.

RC2 40-bit encryptions (RC2En4)

Number of RC2 40-bit cipher encryptions on the NetScaler appliance.

RC2 56-bit encryptions (RC2En5)

Number of RC2 56-bit cipher encryptions on the NetScaler appliance.

RC2 128-bit encryptions (RC2En1)

Number of RC2 128-bit cipher encryptions on the NetScaler appliance.

AES-GCM 128-bit encryptions (AESGCMEn1)

Number of AEC-GCM 128-bit cipher encryptions on the NetScaler appliance.

AES-GCM 256-bit encryptions (AESGCMEn2)

Number of AEC-GCM 256-bit cipher encryptions on the NetScaler appliance.

Null cipher encryptions (NullEn)

Number of Null cipher encryptions on the NetScaler appliance.

MD5 hashes (MD5Hsh)

Number of MD5 hashes on the NetScaler appliance.

SHA hashes (SHAHsh)

Number of SHA hashes on the NetScaler appliance.

SSLv2 SSL handshakes (SSL2Hs)

Number of handshakes on SSLv2 on the NetScaler appliance.

SSLv3 SSL handshakes (SSL3Hs)

Number of handshakes on SSLv3 on the NetScaler appliance.

TLSv1 SSL handshakes (TLS1Hs)

Number of SSL handshakes on TLSv1 on the NetScaler appliance.

TLSv1.1 SSL handshakes (TLS11Hs)

Number of SSL handshakes on TLSv1.1 on the NetScaler appliance.

TLSv1.2 SSL handshakes (TLS12Hs)

Number of SSL handshakes on TLSv1.2 on the NetScaler appliance.

SSLv2 client authentications (SSL2CAt)

Number of client authentications done on SSLv2.

SSLv3 client authentications (SSL3CAt)

Number of client authentications done on SSLv3.

TLSv1 client authentications (TLS1CAt)

Number of client authentications done on TLSv1.

TLSv1.1 client authentications (TLS11CAt)

Number of client authentications done on TLSv1.1.

TLSv1.2 client authentications (TLS12CAt)

Number of client authentications done on TLSv1.2.

RSA authentications (RSAAt)

Number of RSA authentications on the NetScaler appliance.

DH authentications (DHAt)

Number of Diffie-Helman authentications on the NetScaler appliance.

DSS (DSA) authentications (DSSAt)

Total number of times DSS authorization is used on the NetScaler appliance.

Null authentications (NullAt)

Number of Null authentications on the NetScaler appliance.

SSL session renegotiations (BSSLRn)

Number of back-end SSL session renegotiations on the NetScaler appliance.

SSLv3 session renegotiations (BSSL3Rn)

Number of back-end SSLv3 session renegotiations on the NetScaler appliance.

TLsv1 session renegotiations (BTLS1Rn)

Number of back-end TLsv1 session renegotiations on the NetScaler appliance.

TLsv1.1 back-end session renegotiations (BTLS1aRn)

Number of back-end TLsv1.1 session renegotiations on the NetScaler appliance.

TLsv1.2 back-end session renegotiations (BTLS12Rn)

Number of back-end TLsv1.2 session renegotiations on the NetScaler appliance.

RSA 512-bit key exchanges (BRSAX5)

Number of back-end RSA 512-bit key exchanges on the NetScaler appliance.

RSA 1024-bit key exchanges (BRSAX1)

Number of back-end RSA 1024-bit key exchanges on the NetScaler appliance.

RSA 2048-bit key exchanges (BRSAX2)

Number of back-end RSA 2048-bit key exchanges on the NetScaler appliance.

DH 512-bit key exchanges (BDHX5)

Number of back-end DH 512-bit key exchanges on the NetScaler appliance.

DH 1024-bit key exchanges (BDHX1)

Number of back-end DH 1024-bit key exchanges on the NetScaler appliance.

DH 2048-bit key exchanges (BDHX2)

Number of back-end DH 2048-bit key exchanges on the NetScaler appliance.

Backend ECDHE 521 curve key exchanges (BECDEHCx1)

Number of back-end ECDHE 521 curve Key exchanges on the NetScaler appliance.

Backend ECDHE 384 curve key exchanges (BECDEHCx2)

Number of back-end ECDHE 384 curve Key exchanges on the NetScaler appliance.

Backend ECDHE 256 curve key exchanges (BECDEHCx3)

Number of back-end ECDHE 256 curve Key exchanges on the NetScaler appliance.

Backend ECDHE 224 curve key exchanges (BECDEHCx4)

Number of back-end ECDHE 224 curve Key exchanges on the NetScaler appliance.

RC4 40-bit encryptions (BRC4En4)

Number of back-end RC4 40-bit cipher encryptions on the NetScaler appliance.

RC4 56-bit encryptions (BRC4En5)

Number of back-end RC4 56-bit cipher encryptions on the NetScaler appliance.

RC4 64-bit encryptions (BRC4En6)

Number of back-end RC4 64-bit cipher encryptions on the NetScaler appliance.

RC4 128-bit encryptions (BRC4En1)

Number of back-end RC4 128-bit cipher encryptions on the NetScaler appliance.

DES 40-bit encryptions (BDESEn4)

Number of back-end DES 40-bit cipher encryptions on the NetScaler appliance.

DES 56-bit encryptions (BDESEn5)

Number of back-end DES 56-bit cipher encryptions on the NetScaler appliance.

3DES 168-bit encryptions (B3DESE1n)

Number of back-end 3DES 168-bit cipher encryptions on the NetScaler appliance.

AES 128-bit encryptions (BAESEn1)

Back-end AES 128-bit cipher encryptions on the NetScaler appliance.

AES 256-bit encryptions (BAESEn2)

Back-end AES 256-bit cipher encryptions on the NetScaler appliance.

RC2 40-bit encryptions (BRC2En4)

Number of back-end RC2 40-bit cipher encryptions on the NetScaler appliance.

RC2 56-bit encryptions (BRC2En5)

Number of back-end RC2 56-bit cipher encryptions on the NetScaler appliance.

RC2 128-bit encryptions (BRC2En1)

Number of back-end RC2 128-bit cipher encryptions on the NetScaler appliance.

null encryptions (BNullEn)

Number of back-end null cipher encryptions on the NetScaler appliance.

MD5 hashes (BMD5Hsh)

Number of back-end MD5 hashes on the NetScaler appliance.

SHA hashes (BSHAHsh)

Number of back-end SHA hashes on the NetScaler appliance.

SSLv3 handshakes (BSSL3Hs)

Number of back-end SSLv3 handshakes on the NetScaler appliance.

TLSv1 handshakes (BTLS1Hs)

Number of back-end TLSv1 handshakes on the NetScaler appliance.

SSLv3 client authentications (BSSL3CAt)

Number of back-end SSLv3 client authentications on the NetScaler appliance.

TLSv1 client authentications (BTLS1CAt)

Number of back-end TLSv1 client authentications on the NetScaler appliance.

RSA authentications (BRSAAt)

Number of back-end RSA authentications on the NetScaler appliance.

DH authentications (BDHAt)

Number of back-end DH authentications on the NetScaler appliance.

DSS authentications (BDSSAt)

Number of back-end DSS authentications on the NetScaler appliance.

Null authentications (BNullAt)

Number of back-end null authentications on the NetScaler appliance.

RSA key exchanges offloaded (RSAkxOf)

Number of RSA key exchanges offloaded to the cryptography card.

RSA sign operations offloaded (RSASnOf)

Number of RSA sign operations offloaded to the cryptography card.

DH key exchanges offloaded (DHkxOf)

Number of DH key exchanges offloaded to the cryptography card.

RC4 encryptions offloaded (RC4EnOf)

Number of RC4 encryptions offloaded to the cryptography card.

DES encryptions offloaded (DESEnOf)

Number of DES encryptions offloaded to the cryptography card.

AES encryptions offloaded (AESEnOf)

Number of AES encryptions offloaded to the cryptography card.

AES-GCM 128-bit encryptions offloaded (AESGCMEnOf1)

Number of AES-GCM 128-bit encryptions offloaded to the cryptography card.

AES-GCM 256-bit encryptions offloaded (AESGCMEnOf2)

Number of AES-GCM 256-bit encryptions offloaded to the cryptography card.

Bytes encrypted in hardware (EncHw)

Number of bytes encrypted in hardware.

Bytes encrypted in software. (EncSw)

Number of bytes encrypted in software.

Bytes encrypted on the front end. (EncFe)

Number of bytes encrypted on the front end.

Bytes encrypted in hardware on the front end. (EncHwFe)

Number of bytes encrypted in hardware on the front end.

Bytes encrypted in software on front-end (EncSwFe)

Number of bytes encrypted in software on the front end.

Bytes encrypted on back-end (EncBe)

Number of bytes encrypted on the back end.

Bytes encrypted in hardware on back-end (EncHwBe)

Number of bytes encrypted in hardware on the back end.

Bytes encrypted in software on back-end (EncSwBe)

Number of bytes encrypted in software on the back end.

Bytes decrypted in hardware (DecHw)

Number of bytes decrypted in hardware.

Bytes decrypted in software (DecSw)

Number of bytes decrypted in software.

Bytes decrypted on front-end (DecFe)

Number of bytes decrypted on the front end.

Bytes decrypted in hardware on front-end (DecHwFe)

Number of bytes decrypted in hardware on the front end.

Bytes decrypted in software on front-end (DecSwFe)

Number of bytes decrypted in software on the front end.

Bytes decrypted on back-end (DecBe)

Number of bytes decrypted on the back end.

Bytes decrypted in hardware on back-end (DecHwBe)

Number of bytes decrypted in hardware on the back end.

Bytes decrypted in software on the back end. (DecSwBe)

Number of bytes decrypted in software on back-end

Backend SSL sessions reused (BSeRe)

Number of back-end SSL sessions reused on the NetScaler appliance.

IDEA 128-bit encryptions (IDEAEn1)

Number of IDEA 128-bit cipher encryptions on the NetScaler appliance.

IDEA 128-bit encryptions (BIDEAEn1)

Number of back-end IDEA 128-bit cipher encryptions on the NetScaler appliance.

ssl action

The following operations can be performed on "ssl action":

[add](#) | [rm](#) | [show](#)

add ssl action

Creates a new SSL action. An SSL action defines SSL settings that you can apply to the selected requests. You associate an action with one or more policies. Data in client connection requests or responses is compared to a rule (expression) specified in the policy, and the action is applied to connections that match the rule.

Synopsis

```
add ssl action <name> [-clientAuth ( DOCLIENTAUTH | NOCLIENTAUTH )] [-clientCert ( ENABLED | DISABLED ) -certHeader <string>] [-clientCertSerialNumber ( ENABLED | DISABLED ) -certSerialHeader <string>] [-clientCertSubject ( ENABLED | DISABLED ) -certSubjectHeader <string>] [-clientCertHash ( ENABLED | DISABLED ) -certHashHeader <string>] [-clientCertIssuer ( ENABLED | DISABLED ) -certIssuerHeader <string>] [-sessionID ( ENABLED | DISABLED ) -sessionIDHeader <string>] [-cipher ( ENABLED | DISABLED ) -cipherHeader <string>] [-clientCertNotBefore ( ENABLED | DISABLED ) -certNotBeforeHeader <string>] [-clientCertNotAfter ( ENABLED | DISABLED ) -certNotAfterHeader <string>] [-OWASupport ( ENABLED | DISABLED )]
```

Arguments

name

Name for the SSL action. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the action is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my action" or 'my action').

clientAuth

Perform client certificate authentication.

Possible values: DOCLIENTAUTH, NOCLIENTAUTH

clientCert

Insert the entire client certificate into the HTTP header of the request being sent to the web server. The certificate is inserted in ASCII (PEM) format.

Possible values: ENABLED, DISABLED

certHeader

Name of the header into which to insert the client certificate.

clientCertSerialNumber

Insert the entire client serial number into the HTTP header of the request being sent to the web server.

Possible values: ENABLED, DISABLED

certSerialHeader

Name of the header into which to insert the client serial number.

clientCertSubject

Insert the client certificate subject, also known as the distinguished name (DN), into the HTTP header of the request being sent to the web server.

Possible values: ENABLED, DISABLED

certSubjectHeader

Name of the header into which to insert the client certificate subject.

clientCertHash

Insert the certificate signature (hash) into the HTTP header of the request being sent to the web server.

Possible values: ENABLED, DISABLED

certHashHeader

Name of the header into which to insert the client certificate signature (hash).

clientCertIssuer

Insert the certificate issuer details into the HTTP header of the request being sent to the web server.

Possible values: ENABLED, DISABLED

certIssuerHeader

Name of the header into which to insert the client certificate issuer details.

sessionID

Insert the SSL session ID into the HTTP header of the request being sent to the web server. Every SSL connection that the client and the NetScaler share has a unique ID that identifies the specific connection.

Possible values: ENABLED, DISABLED

sessionIDHeader

Name of the header into which to insert the Session ID.

cipher

Insert the cipher suite that the client and the NetScaler appliance negotiated for the SSL session into the HTTP header of the request being sent to the web server. The appliance inserts the cipher-suite name, SSL protocol, export or non-export string, and cipher strength bit, depending on the type of browser connecting to the SSL virtual server or service (for example, Cipher-Suite: RC4- MD5 SSLv3 Non-Export 128-bit).

Possible values: ENABLED, DISABLED

cipherHeader

Name of the header into which to insert the name of the cipher suite.

clientCertNotBefore

Insert the date from which the certificate is valid into the HTTP header of the request being sent to the web server. Every certificate is configured with the date and time from which it is valid.

Possible values: ENABLED, DISABLED

certNotBeforeHeader

Name of the header into which to insert the date and time from which the certificate is valid.

clientCertNotAfter

Insert the date of expiry of the certificate into the HTTP header of the request being sent to the web server. Every certificate is configured with the date and time at which the certificate expires.

Possible values: ENABLED, DISABLED

certNotAfterHeader

Name of the header into which to insert the certificate's expiry date.

OWASupport

If the appliance is in front of an Outlook Web Access (OWA) server, insert a special header field, FRONT-END-HTTPS: ON, into the HTTP requests going to the OWA server. This header communicates to the server that the transaction is HTTPS and not HTTP.

Possible values: ENABLED, DISABLED

Example

```
add ssl action certInsert_act -clientCert ENABLED -certHeader CERT
```

rm ssl action

Removes the specified SSL action.

Synopsys

```
rm ssl action <name>
```

Arguments

name

Name of the SSL action to remove.

Example

```
rm ssl action certInsert_act
```

show ssl action

Displays information about all the SSL actions configured on the appliance, or displays detailed information about the specified SSL action.

Synopsys

```
show ssl action [<name>]
```

Arguments

name

Name of the SSL action for which to show detailed information.

Outputs

stateflag

clientAuth

Perform client certificate authentication.

clientCert

Insert the entire client certificate into the HTTP header of the request being sent to the web server. The certificate is inserted in ASCII (PEM) format.

certHeader

clientCertSerialNumber

Insert the entire client serial number into the HTTP header of the request being sent to the web server.

certSerialHeader

clientCertSubject

Insert the client certificate subject, also known as the distinguished name (DN), into the HTTP header of the request being sent to the web server.

certSubjectHeader**clientCertHash**

Insert the certificate signature (hash) into the HTTP header of the request being sent to the web server.

certHashHeader**clientCertIssuer**

Insert the certificate issuer details into the HTTP header of the request being sent to the web server.

certIssuerHeader**sessionID**

Insert the SSL session ID into the HTTP header of the request being sent to the web server. Every SSL connection that the client and the NetScaler share has a unique ID that identifies the specific connection.

sessionIDHeader**cipher**

Insert the cipher suite that the client and the NetScaler appliance negotiated for the SSL session into the HTTP header of the request being sent to the web server. The appliance inserts the cipher-suite name, SSL protocol, export or non-export string, and cipher strength bit, depending on the type of browser connecting to the SSL virtual server or service (for example, Cipher-Suite: RC4- MD5 SSLv3 Non-Export 128-bit).

cipherHeader**OWASupport**

If the appliance is in front of an Outlook Web Access (OWA) server, insert a special header field, FRONT-END-HTTPS: ON, into the HTTP requests going to the OWA server. This header communicates to the server that the transaction is HTTPS and not HTTP.

clientCertNotBefore

Insert the date from which the certificate is valid into the HTTP header of the request being sent to the web server. Every certificate is configured with the date and time from which it is valid.

certNotBeforeHeader**clientCertNotAfter**

Insert the date of expiry of the certificate into the HTTP header of the request being sent to the web server. Every certificate is configured with the date and time at which the certificate expires.

certNotAfterHeader**hits**

The number of times the action has been taken.

undefHits

The number of times the action resulted in UNDEF.

referenceCount

The number of references to the action.

description

Description of the action

flags

builtin

Flag to determine whether ssl action is built-in or not

devno

count

Example

```
show ssl action 1 Configured SSL action: 1)      Name: certInsert_act      Data Insert:
```

ssl cert

The following operations can be performed on "ssl cert":

create ssl cert

Generates a signed X509 Certificate.

Synopsys

```
create ssl cert <certFile> <reqFile> <certType> [-keyFile <input_filename>] [-keyform ( DER | PEM ) {-PEMPassPhrase }] [-days <positive_integer>] [-certForm ( DER | PEM )] [-CAcert <input_filename>] [-CAcertForm ( DER | PEM )] [-CAkey <input_filename>] [-CAkeyForm ( DER | PEM )] [-CAserial <output_filename>]
```

Arguments

certFile

Name for and, optionally, path to the generated certificate file. /nsconfig/ssl/ is the default path.

Maximum value: 63

reqFile

Name for and, optionally, path to the certificate-signing request (CSR). /nsconfig/ssl/ is the default path.

Maximum value: 63

certType

Type of certificate to generate. Specify one of the following:

- * ROOT_CERT - Self-signed Root-CA certificate. You must specify the key file name. The generated Root-CA certificate can be used for signing end-user client or server certificates or to create Intermediate-CA certificates.

- * INTM_CERT - Intermediate-CA certificate.

- * CLNT_CERT - End-user client certificate used for client authentication.

- * SRVR_CERT - SSL server certificate used on SSL servers for end-to-end encryption.

Possible values: ROOT_CERT, INTM_CERT, CLNT_CERT, SRVR_CERT

keyFile

Name for and, optionally, path to the private key. You can either use an existing RSA or DSA key that you own or create a new private key on the NetScaler appliance. This file is required only when creating a self-signed Root-CA certificate. The key file is stored in the /nsconfig/ssl directory by default.

If the input key specified is an encrypted key, you are prompted to enter the PEM pass phrase that was used for encrypting the key.

Maximum value: 63

keyform

Format in which the key is stored on the appliance.

Possible values: DER, PEM

Default value: PEM

PEMPassPhrase

days

Number of days for which the certificate will be valid, beginning with the time and day (system time) of creation.

Default value: 365

Minimum value: 1

Maximum value: 3650

certForm

Format in which the certificate is stored on the appliance.

Possible values: DER, PEM

Default value: PEM

CAcert

Name of the CA certificate file that issues and signs the Intermediate-CA certificate or the end-user client and server certificates.

Maximum value: 63

CAcertForm

Format of the CA certificate.

Possible values: DER, PEM

Default value: PEM

CAkey

Private key, associated with the CA certificate that is used to sign the Intermediate-CA certificate or the end-user client and server certificate. If the CA key file is password protected, the user is prompted to enter the pass phrase that was used to encrypt the key.

Maximum value: 63

CAkeyForm

Format for the CA certificate.

Possible values: DER, PEM

Default value: PEM

CAserial

Serial number file maintained for the CA certificate. This file contains the serial number of the next certificate to be issued or signed by the CA. If the specified file does not exist, a new file is created, with /nsconfig/ssl/ as the default path. If you do not specify a proper path for the existing serial file, a new serial file is created. This might change the certificate serial numbers assigned by the CA certificate to each of the certificates it signs.

Maximum value: 63

Example

```
1) create ssl cert /nsconfig/ssl/root_cert.pem /nsconfig/ssl/root_csr.pem ROOT_CERT -keyF:
```


ssl certChain

The following operations can be performed on "ssl certChain":

show ssl certChain

Display all the certificates attached to this particular certificate.

Synopsys

```
show ssl certChain [<CertKeyName>]
```

Arguments

CertKeyName

Name of the Certificate

Outputs

linkCertKeyName

Name of the Linked Certificate

IsLinked

Used to find if certificate is linked

IsCA

Used to find if certificate is a CA

AddSubject

Used to find if certificate is linked

stateflag

devno

count

Example

```
show certchain [certificate name]
```

ssl certFile

The following operations can be performed on "ssl certFile":

[import](#) | [rm](#) | [show](#)

import ssl certFile

Imports a certificate file to the NetScaler appliance, assigns it a name, and stores it in the /nsconfig/ssl/certfile folder. The folder is created if it does not exist.

Synopsys

```
import ssl certFile <name> <src>
```

Arguments

name

Name to assign to the imported certificate file. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. The following requirement applies only to the NetScaler CLI: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my file" or 'my file').

src

URL specifying the protocol, host, and path, including file name, to the certificate file to be imported. For example, http://www.example.com/cert_file.

NOTE: The import fails if the object to be imported is on an HTTPS server that requires client certificate authentication for access.

Example

```
import ssl certfile my-certfile http://www.example.com/cert_file
```

rm ssl certFile

Deletes the specified certificate file.

Synopsys

```
rm ssl certFile <name>
```

Arguments

name

Name of the certificate file to delete.

Example

```
rm ssl certfile my-certfile
```

show ssl certFile

Displays lists of all the imported certificate file objects on the NetScaler ADC.

Synopsys

```
show ssl certFile
```

Outputs

name

Name to assign to the imported certificate file. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. The following requirement applies only to the NetScaler CLI: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my file" or 'my file').

src

URL specifying the protocol, host, and path, including file name, to the certificate file to be imported. For example, `http://www.example.com/cert_file`.

NOTE: The import fails if the object to be imported is on an HTTPS server that requires client certificate authentication for access.

devno

count

stateflag

Example

```
show ssl certfile
```

ssl certKey

The following operations can be performed on "ssl certKey":

add | **rm** | **set** | **unset** | **bind** | **unbind** | **link** | **unlink** | **show** | **update**

add ssl certKey

Adds a certificate-key pair to memory. After it is bound to a virtual server or service, it is used for processing SSL transactions. In a high-availability configuration, the path to the certificate and the optional private key must be the same on the primary and the secondary appliance. For a server certificate, a private key is required.

Synopsys

```
add ssl certKey <certkeyName> -cert <string> [(-key <string> [-password]) | -fipsKey <string>] [-inform ( DER | PEM )] [-expiryMonitor ( ENABLED | DISABLED ) [-notificationPeriod <positive_integer>]] [-bundle ( YES | NO )]
```

Arguments

certkeyName

Name for the certificate and private-key pair. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the certificate-key pair is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my cert" or 'my cert').

cert

Name of and, optionally, path to the X509 certificate file that is used to form the certificate-key pair. The certificate file should be present on the appliance's hard-disk drive or solid-state drive. Storing a certificate in any location other than the default might cause inconsistency in a high availability setup. /nsconfig/ssl/ is the default path.

key

Name of and, optionally, path to the private-key file that is used to form the certificate-key pair. The certificate file should be present on the appliance's hard-disk drive or solid-state drive. Storing a certificate in any location other than the default might cause inconsistency in a high availability setup. /nsconfig/ssl/ is the default path.

password

Passphrase that was used to encrypt the private-key. Use this option to load encrypted private-keys in PEM format.

fipsKey

Name of the FIPS key that was created inside the Hardware Security Module (HSM) of a FIPS appliance, or a key that was imported into the HSM.

inform

Input format of the certificate and the private-key files. The two formats supported by the appliance are:

PEM - Privacy Enhanced Mail

DER - Distinguished Encoding Rule

Possible values: DER, PEM

Default value: PEM

passplain

Pass phrase used to encrypt the private-key. Required when adding an encrypted private-key in PEM format.

expiryMonitor

Issue an alert when the certificate is about to expire.

Possible values: ENABLED, DISABLED

notificationPeriod

Time, in number of days, before certificate expiration, at which to generate an alert that the certificate is about to expire.

Minimum value: 10

Maximum value: 100

bundle

Parse the certificate chain as a single file after linking the server certificate to its issuer's certificate within the file.

Possible values: YES, NO

Default value: NO

Example

```
1) add ssl certkey siteAcertkey -cert /nsconfig/ssl/cert.pem -key /nsconfig/ssl/pkey.pem
```

rm ssl certKey

Removes all the certificate-key pairs, or the specified certificate-key pair, from the appliance. The certificate-key pair is removed only if it is not referenced by any other object. The reference count is updated when the certificate-key pair is bound to an SSL virtual server or linked to another certificate-key pair.

Synopsys

```
rm ssl certKey <certkeyName> ...
```

Arguments

certkeyName

Name of the certificate-key pair to remove.

Example

```
1) rm ssl certkey siteAcertkey The above command removes the certificate-key pair siteAce:
```

set ssl certKey

Modifies the specified attributes of a certificate-key pair.

Synopsys

```
set ssl certKey <certkeyName> [-expiryMonitor ( ENABLED | DISABLED ) [-notificationPeriod <positive_integer>]]
```

Arguments

certkeyName

Name of the certificate-key pair to modify.

expiryMonitor

Issue an alert when the certificate is about to expire.

Possible values: ENABLED, DISABLED

notificationPeriod

Time, in number of days, before certificate expiration, at which to generate an alert that the certificate is about to expire.

Minimum value: 10

Maximum value: 100

unset ssl certKey

Use this command to remove ssl certKey settings. Refer to the set ssl certKey command for meanings of the arguments.

Synopsis

```
unset ssl certKey <certkeyName> [-expiryMonitor] [-notificationPeriod]
```

bind ssl certKey

Binds a certificate-key pair to an SSL virtual server or an SSL service.

Synopsis

```
bind ssl certKey [<certkeyName>] [-ocspResponder <string>] [-priority <positive_integer>]
```

Arguments

certkeyName

Name of the certificate-key pair.

ocspResponder

Name of the OCSP responder to be associated with the CA certificate.

priority

Priority of the OCSP responder binding.

Minimum value: 1

Maximum value: 32

Example

```
1) bind ssl certkey cacert -ocspResponder ocsp_ca -priority 1 In the above example, the C
```

unbind ssl certKey

Unbinds the specified certificate-key pair from the SSL virtual server or service.

Synopsis

```
unbind ssl certKey <certkeyName> -ocspResponder <string>
```

Arguments

certkeyName

Name of the certificate-key pair to unbind.

ocspResponder

Name of the OCSP responder.

Example

```
1) unbind ssl certkey sslvip siteAcertkey In the above example, the server certificate si
```

link ssl certKey

Links a certificate-key pair to its Certificate Authority (CA) certificate-key pair.

Synopsys

```
link ssl certKey <certkeyName> <linkCertKeyName>
```

Arguments

certkeyName

Name of the certificate-key pair to link to its issuer's certificate-key pair in the chain.

linkCertKeyName

Name of the Certificate Authority certificate-key pair to which to link a certificate-key pair.

Example

```
1) link ssl certkey siteAcertkey CAcertkey In the above example, the certificate-key site:
```

unlink ssl certKey

Unlinks the certificate-key pair from its Certificate-Authority (CA) certificate-key pair.

Synopsys

```
unlink ssl certKey <certkeyName>
```

Arguments

certkeyName

Name of the certificate-key pair to unlink.

Example

```
1) unlink ssl certkey siteAcertkey The above example unlinks the certificate 'siteAcertkey'
```

show ssl certKey

Displays information about all the certificate-key pairs configured on the appliance, or displays detailed information about the specified certificate-key pair.

Synopsys

```
show ssl certKey [<certkeyName>]
```

Arguments

certkeyName

Name of the certificate-key pair for which to show detailed information.

Outputs

cert

The name and location of the file containing the certificate.

key

The name and location of the file containing the key.

inform

The encoding format of the certificate and key (PEM or DER).

signatureAlg

Signature algorithm.

serial

Serial number.

issuer

Issuer name.

clientCertNotBefore

Not-Before date.

clientCertNotAfter

Not-After date.

daysToExpiration

Days remaining for the certificate to expire.

subject

Subject name.

publickey

Public key algorithm.

publickeysize

Size of the public key.

version

Version.

priority

ocsp priority

status

Status of the certificate.

fipsKey

FIPS key ID.

passcrypt

Passcrypt.

data

Vserver Id

serverName

Vserver name to which the certificate key pair is bound.

serviceName

Service name to which the certificate key pair is bound.

ocspResponder

OCSP responders bound to this certkey

expiryMonitor

Certificate expiry monitor

notificationPeriod

Certificate expiry notification period

linkCertKeyName

The name of the Certificate-Authority.

stateflag**devno****count**

Example

1) An example of the output of the `show ssl certkey` command is shown below: 2 configured

update ssl certKey

Updates the certificate or private key in a certificate-key pair. In a high availability configuration, the path to the certificate and the optional private key must be the same on the primary and secondary nodes.

Synopsys

```
update ssl certKey <certkeyName> [-cert <string>] [(-key <string> [-password]) | -fipsKey <string>] [-inform ( DER | PEM )] [-noDomainCheck]
```

Arguments

certkeyName

Name of the certificate-key pair to update.

cert

Name of and, optionally, path to the X509 certificate file that is used to form the certificate-key pair. The certificate file should be present on the appliance's hard-disk drive or solid-state drive. Storing a certificate in any location other than the default might cause inconsistency in a high availability setup. `/nsconfig/ssl/` is the default path.

key

Name of and, optionally, path to the private-key file that is used to form the certificate-key pair. The certificate file should be present on the appliance's hard-disk drive or solid-state drive. Storing a certificate in any location other than the default might cause inconsistency in a high availability setup. `/nsconfig/ssl/` is the default path.

password

Passphrase that was used to encrypt the private-key. Use this option to load encrypted private-keys in PEM format.

fipsKey

Name of the FIPS key that was created inside the Hardware Security Module (HSM) of a FIPS appliance, or a key that was imported into the HSM.

inform

Input format of the certificate and the private-key files. The two formats supported by the appliance are:

PEM - Privacy Enhanced Mail

DER - Distinguished Encoding Rule

Possible values: DER, PEM

Default value: PEM

passplain

Pass phrase used to encrypt the private-key. Required when adding an encrypted private-key in PEM format.

noDomainCheck

Override the check for matching domain names during a certificate update operation.

Example

```
1)      update ssl certkey siteAcertkey -cert /nsconfig/ssl/cert.pem -key /nsconfig/ssl/pk
```

ssl certLink

The following operations can be performed on "ssl certLink":

show ssl certLink

Displays information about all the linked certificate-key pairs on the appliance.

Synopsys

```
show ssl certLink
```

Outputs

certkeyName

Certificate key name.

linkCertKeyName

Name of the Certificate-Authority.

devno

count

stateflag

Example

The following shows an example of the output of the show ssl certlink command: linked ceri

ssl certReq

The following operations can be performed on "ssl certReq":

create ssl certReq

Generates a new Certificate Signing Request (CSR). A CSR is a collection of information including the domain name, company details, and the private key to be used to create the certificate. Send the CSR to a Certificate Authority (CA) to obtain an X509 certificate for the user domain (web site).

Synopsys

```
create ssl certReq <reqFile> (-keyFile <input_filename> | -fipsKeyName <string>) [-keyform ( DER | PEM ) {-PEMPassPhrase }] -countryName <string> -stateName <string> -organizationName <string> [-organizationUnitName <string>] [-localityName <string>] [-commonName <string>] [-emailAddress <string>] [-challengePassword ] [-companyName <string>]
```

Arguments

reqFile

Name for and, optionally, path to the certificate signing request (CSR). /nsconfig/ssl/ is the default path.

Maximum value: 63

keyFile

Name of and, optionally, path to the private key used to create the certificate signing request, which then becomes part of the certificate-key pair. The private key can be either an RSA or a DSA key. The key must be present in the appliance's local storage. /nsconfig/ssl is the default path.

Maximum value: 63

fipsKeyName

Name of the FIPS key used to create the certificate signing request. FIPS keys are created inside the Hardware Security Module of the FIPS card.

keyform

Format in which the key is stored on the appliance.

Possible values: DER, PEM

Default value: PEM

PEMPassPhrase

countryName

Two letter ISO code for your country. For example, US for United States.

stateName

Full name of the state or province where your organization is located.

Do not abbreviate.

organizationName

Name of the organization that will use this certificate. The organization name (corporation, limited partnership, university, or government agency) must be registered with some authority at the national, state, or city level. Use the legal name under which the organization is registered.

Do not abbreviate the organization name and do not use the following characters in the name:

Angle brackets (< >) tilde (~), exclamation mark, at (@), pound (#), zero (0), caret (^), asterisk (*), forward slash (/), square brackets ([]), question mark (?).

organizationUnitName

Name of the division or section in the organization that will use the certificate.

localityName

Name of the city or town in which your organization's head office is located.

commonName

Fully qualified domain name for the company or web site. The common name must match the name used by DNS servers to do a DNS lookup of your server. Most browsers use this information for authenticating the server's certificate during the SSL handshake. If the server name in the URL does not match the common name as given in the server certificate, the browser terminates the SSL handshake or prompts the user with a warning message.

Do not use wildcard characters, such as asterisk (*) or question mark (?), and do not use an IP address as the common name. The common name must not contain the protocol specifier <http://> or <https://>.

emailAddress

Contact person's e-mail address. This address is publically displayed as part of the certificate. Provide an e-mail address that is monitored by an administrator who can be contacted about the certificate.

challengePassword

Pass phrase, embedded in the certificate signing request that is shared only between the client or server requesting the certificate and the SSL certificate issuer (typically the certificate authority). This pass phrase can be used to authenticate a client or server that is requesting a certificate from the certificate authority.

companyName

Additional name for the company or web site.

Example

```
create ssl certreq /nsconfig/ssl/csr.pem -keyFile /nsconfig/ssl/rsa1024.pem
```

ssl cipher

The following operations can be performed on "ssl cipher":

add | **bind** | **show** | **rm** | **unbind**

add ssl cipher

Creates a user-defined cipher group, which you can bind to an SSL virtual server instead of binding ciphers individually. Although you cannot modify a built-in cipher group, you can add built-in cipher groups as well as individual ciphers to a user-defined cipher group.

Synopsis

```
add ssl cipher <cipherGroupName>
```

Arguments

cipherGroupName

Name for the user-defined cipher group. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the cipher group is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my ciphergroup" or 'my ciphergroup').

Example

```
1) add ssl cipher mygroup SSL2-RC4-MD5 SSL2-EXP-RC4-MD5 The above command creates a new c
```

bind ssl cipher

Adds ciphers to a user-defined cipher group. You can add an existing cipher group to a user-defined cipher group but you cannot modify a built-in cipher group.

Synopsis

```
bind ssl cipher [<cipherGroupName>@] [-cipherName <string>]
```

Arguments

cipherGroupName

Name of the user-defined cipher group.

cipherName

Name of the individual cipher, user-defined cipher group, or predefined (built-in) cipher alias to add to the cipher group.

Example

```
1) bind ssl cipher sslvip ADD SSL3-RC4-SHA The above example appends the cipher SSL3-RC4-S
```

show ssl cipher

Displays information about all the cipher groups defined on the appliance, or displays detailed information about the specified cipher group.

Synopsys

show ssl cipher [<cipherGroupName>]

Arguments

cipherGroupName

Name of the cipher group for which to show detailed information.

Outputs

description

Cipher suite description.

cipherName

Cipher name.

flag

stateflag

peFlags

devno

count

Example

1) An example of the output of the show ssl cipher SSL3-RC4-MD5 command is as follows: Ci

rm ssl cipher

Removes a user-defined cipher group from the appliance.

Synopsys

rm ssl cipher <cipherGroupName>

Arguments

cipherGroupName

Name of the user-defined cipher group to remove.

Example

1) rm ssl cipher mygroup SSL2-RC4-MD5 The above example removes the cipher SSL2-RC4-MD5 f

unbind ssl cipher

Removes all the ciphers from a user-defined cipher group. You can only remove individual ciphers from a user-defined cipher group. Removing groups is not supported.

Synopsys

unbind ssl cipher <cipherGroupName> [-cipherName <string> ...]

Arguments

cipherGroupName

Name of the user-defined cipher group.

cipherName

Name(s) of the cipher(s) to be removed from the user-defined cipher group.

Example

1) `rm ssl cipher mygroup SSL2-RC4-MD5` The above example removes the cipher SSL2-RC4-MD5 from the user-defined cipher group mygroup.

ssl ciphersuite

The following operations can be performed on "ssl ciphersuite":

show ssl ciphersuite

Displays information about all the cipher suites configured on the appliance, or displays detailed information about the specified cipher-suite. A cipher suite comprises a protocol and the following algorithms: key exchange (Kx), authentication (Au), encryption (Enc), and message authentication code (Mac).

Synopsys

```
show ssl ciphersuite [<cipherName>]
```

Arguments

cipherName

Name of the cipher suite for which to show detailed information.

Outputs

description

Cipher suite description.

flag

stateflag

devno

count

Example

1) An example of the output of the `show ssl cipher SSL3-RC4-MD5` command is as follows: Cij

ssl crl

The following operations can be performed on "ssl crl":

add | **create** | **rm** | **set** | **unset** | **show**

add ssl crl

Adds a Certificate Revocation List (CRL). A CRL identifies invalid certificates by serial number and issuer. In a high availability configuration, the CRL must be in the same location on the primary and secondary nodes.

Synopsis

```
add ssl crl <crlName> <crlPath> [-inform ( DER | PEM )] [-refresh ( ENABLED | DISABLED )] [-CAcert <string>] [-method ( HTTP | LDAP )] [-server <ip_addr|ipv6_addr|*> | -url <URL>] [-port <port>] [-baseDN <string>] [-scope ( Base | One )] [-interval <interval>] [-day <integer>] [-time <HH:MM>] [-bindDN <string>] {-password } [-binary ( YES | NO )]
```

Arguments

crlName

Name for the Certificate Revocation List (CRL). Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the CRL is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my crl" or 'my crl').

crlPath

Path to the CRL file. /var/netscaler/ssl/ is the default path.

inform

Input format of the CRL file. The two formats supported on the appliance are:

PEM - Privacy Enhanced Mail.

DER - Distinguished Encoding Rule.

Possible values: DER, PEM

Default value: PEM

refresh

Set CRL auto refresh.

Possible values: ENABLED, DISABLED

CAcert

CA certificate that has issued the CRL. Required if CRL Auto Refresh is selected. Install the CA certificate on the appliance before adding the CRL.

method

Method for CRL refresh. If LDAP is selected, specify the method, CA certificate, base DN, port, and LDAP server name. If HTTP is selected, specify the CA certificate, method, URL, and port. Cannot be changed after a CRL is added.

Possible values: HTTP, LDAP

server

IP address of the LDAP server from which to fetch the CRLs.

url

URL of the CRL distribution point.

port

Port for the LDAP server.

Minimum value: 1

baseDN

Base distinguished name (DN), which is used in an LDAP search to search for a CRL. Citrix recommends searching for the Base DN instead of the Issuer Name from the CA certificate, because the Issuer Name field might not exactly match the LDAP directory structure's DN.

scope

Extent of the search operation on the LDAP server. Available settings function as follows:

One - One level below Base DN.

Base - Exactly the same level as Base DN.

Possible values: Base, One

Default value: One

interval

CRL refresh interval. Use the NONE setting to unset this parameter.

Possible values: MONTHLY, WEEKLY, DAILY, NONE

day

Day on which to refresh the CRL, or, if the Interval parameter is not set, the number of days after which to refresh the CRL. If Interval is set to MONTHLY, specify the date. If Interval is set to WEEKLY, specify the day of the week (for example, Sun=0 and Sat=6). This parameter is not applicable if the Interval is set to DAILY.

Maximum value: 31

time

Time, in hours (1-24) and minutes (1-60), at which to refresh the CRL.

bindDN

Bind distinguished name (DN) to be used to access the CRL object in the LDAP repository if access to the LDAP repository is restricted or anonymous access is not allowed.

password

Password to access the CRL in the LDAP repository if access to the LDAP repository is restricted or anonymous access is not allowed.

binary

Set the LDAP-based CRL retrieval mode to binary.

Possible values: YES, NO

Default value: NO

Example

```
1) add ssl certkey CACert -cert /nsconfig/ssl/ca_cert.pem add ssl crl crl_file /var/netsc:
```

create ssl crl

Revokes a certificate, or list of certificates, or generates a CRL for the list of revoked certificates.

Synopsys

```
create ssl crl <CAcertFile> <CAkeyFile> <indexFile> (-revoke <input_filename> | -genCRL <output_filename>) {-password }
```

Arguments

CAcertFile

Name of and, optionally, path to the CA certificate file.

/nsconfig/ssl/ is the default path.

Maximum value: 63

CAkeyFile

Name of and, optionally, path to the CA key file. /nsconfig/ssl/ is the default path

Maximum value: 63

indexFile

Name of and, optionally, path to the file containing the serial numbers of all the certificates that are revoked. Revoked certificates are appended to the file. /nsconfig/ssl/ is the default path

Maximum value: 63

revoke

Name of and, optionally, path to the certificate to be revoked. /nsconfig/ssl/ is the default path.

Maximum value: 63

genCRL

Name of and, optionally, path to the CRL file to be generated. The list of certificates that have been revoked is obtained from the index file. /nsconfig/ssl/ is the default path.

Maximum value: 63

password

Password for the CA key file.

Maximum value: 31

Example

```
1) create crl /nsconfig/ssl/cacert.pem /nsconfig/ssl/cakey.pem /nsconfig/ssl/index.txt -g
```

rm ssl crl

Removes the specified CRL from the appliance.

Synopsys

```
rm ssl crl <crlName> ...
```

Arguments

crlName

Name of the CRL to remove.

Example

```
1) rm ssl crl ca_crl
```

 The above CLI command to delete the CRL object ca_crl from the system

set ssl crl

Modifies all the parameters of a CRL, except the CRL name and method.

Synopsys

```
set ssl crl <crlName> [-refresh ( ENABLED | DISABLED )] [-CAcert <string>] [-server <ip_addr|ipv6_addr*> | -url <URL>] [-method ( HTTP | LDAP )] [-port <port>] [-baseDN <string>] [-scope ( Base | One )] [-interval <interval>] [-day <integer>] [-time <HH:MM>] [-bindDN <string>] {-password } [-binary ( YES | NO )]
```

Arguments

crlName

Name of the CRL to be modified.

refresh

Set CRL auto refresh.

Possible values: ENABLED, DISABLED

CAcert

CA certificate that has issued the CRL. Required if CRL Auto Refresh is selected. Install the CA certificate on the appliance before adding the CRL.

server

IP address of the LDAP server from which to fetch the CRLs.

method

Method for CRL refresh. If LDAP is selected, specify the method, CA certificate, base DN, port, and LDAP server name. If HTTP is selected, specify the CA certificate, method, URL, and port. Cannot be changed after a CRL is added.

Possible values: HTTP, LDAP

url

URL of the CRL distribution point.

port

Port for the LDAP server.

Minimum value: 1

baseDN

Base distinguished name (DN), which is used in an LDAP search to search for a CRL. Citrix recommends searching for the Base DN instead of the Issuer Name from the CA certificate, because the Issuer Name field might not exactly match the LDAP directory structure's DN.

scope

Extent of the search operation on the LDAP server. Available settings function as follows:

One - One level below Base DN.

Base - Exactly the same level as Base DN.

Possible values: Base, One

Default value: One

interval

CRL refresh interval. Use the NONE setting to unset this parameter.

Possible values: MONTHLY, WEEKLY, DAILY, NOW, NONE

day

Day on which to refresh the CRL, or, if the Interval parameter is not set, the number of days after which to refresh the CRL. If Interval is set to MONTHLY, specify the date. If Interval is set to WEEKLY, specify the day of the week (for example, Sun=0 and Sat=6). This parameter is not applicable if the Interval is set to DAILY.

Maximum value: 31

time

Time, in hours (1-24) and minutes (1-60), at which to refresh the CRL.

bindDN

Bind distinguished name (DN) to be used to access the CRL object in the LDAP repository if access to the LDAP repository is restricted or anonymous access is not allowed.

password

Password to access the CRL in the LDAP repository if access to the LDAP repository is restricted or anonymous access is not allowed.

binary

Set the LDAP-based CRL retrieval mode to binary.

Possible values: YES, NO

Default value: NO

Example

```
1) set ssl crl crl_file -refresh ENABLE -interval MONTHLY -days 10 -time 12:00 The above c
```

unset ssl crl

Use this command to remove ssl crl settings. Refer to the set ssl crl command for meanings of the arguments.

Synopsys

```
unset ssl crl <crlName> [-refresh] [-CAcert] [-server] [-method] [-url] [-port] [-baseDN] [-scope] [-interval] [-day] [-time]  
[-bindDN] [-password] [-binary]
```

show ssl crl

Displays information about all the CRLs configured on the appliance, or displays detailed information about the specified CRL.

Synopsys

```
show ssl crl [<crlName>]
```

Arguments

crlName

Name of the CRL for which to show detailed information.

Outputs

crlPath

The name and path to the file containing the CRL.

inform

The encoding format of the CRL (PEM or DER).

CAcert

The CA certificate that issued the CRL.

refresh

The state of the auto refresh feature for the CRL.

scope

Extent of the search operation on the LDAP server.

Base: Exactly the same level as basedn

One : One level below basedn.

server

The IP address of the LDAP/HTTP server from which the CRLs are to be fetched.

port

The port of the LDAP/HTTP server.

url

URL of the CRL distribution point.

method

The method for CRL refresh (LDAP or HTTP).

baseDN

The baseDN to be used to fetch the CRL object from the LDAP server.

interval

The CRL refresh interval.

day

The day when the CRL is to be refreshed.

time

The time when the CRL is to be refreshed.

bindDN

The bindDN to be used to access the CRL object in the LDAP repository.

password

The password to be is used to access the CRL object in the LDAP repository.

flags

CRL status flag.

lastupdatetime

Last CRL refresh time.

version

CRL version.

signaturealgo

Signature algorithm.

issuer

Issuer name.

lastupdate

Last update time.

nextupdate

Next update time.

date

Certificate Revocation date

number

Certificate Serial number.

binary

Mode of retrieval of CRL from LDAP server.

daysToExpiration

Number of days remaining for the CRL to expire.

devno**count****stateflag**

Example

1) An example output of the `show ssl crl` command is as follows: 1 configured CRL(s) 1 Name

ssl crlFile

The following operations can be performed on "ssl crlFile":

[import](#) | [rm](#) | [show](#)

import ssl crlFile

Imports a CRL file to the NetScaler appliance, assigns it a name, and stores it in the /var/netscaler/ssl/crlfile folder. The folder is created if it does not exist.

Synopsys

```
import ssl crlFile <name> <src>
```

Arguments

name

Name to assign to the imported CRL file. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. The following requirement applies only to the NetScaler CLI: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my file" or 'my file').

src

URL specifying the protocol, host, and path, including file name to the CRL file to be imported. For example, http://www.example.com/crl_file.

NOTE: The import fails if the object to be imported is on an HTTPS server that requires client certificate authentication for access.

Example

```
import ssl crlfile my-crlfile http://www.example.com/crl_file
```

rm ssl crlFile

Deletes the specified CRL file.

Synopsys

```
rm ssl crlFile <name>
```

Arguments

name

Name of the CRL file to delete.

Example

```
rm ssl crlfile my-crlfile
```

show ssl crlFile

Displays lists of all the imported CRL file objects on the NetScaler ADC.

Synopsys

```
show ssl crlFile
```

Outputs

name

Name to assign to the imported CRL file. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. The following requirement applies only to the NetScaler CLI: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my file" or 'my file').

src

URL specifying the protocol, host, and path, including file name to the CRL file to be imported. For example, `http://www.example.com/crl_file`.

NOTE: The import fails if the object to be imported is on an HTTPS server that requires client certificate authentication for access.

devno

count

stateflag

Example

```
show ssl crlfile
```

ssl dhFile

The following operations can be performed on "ssl dhFile":

[import](#) | [rm](#) | [show](#)

import ssl dhFile

Imports a DH file to the NetScaler appliance, assigns it a name, and stores it in the /nsconfig/ssl/dhfile folder. The folder is created if it does not exist.

Synopsys

```
import ssl dhFile <name> <src>
```

Arguments

name

Name to assign to the imported DH file. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. The following requirement applies only to the NetScaler CLI: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my file" or 'my file').

src

URL specifying the protocol, host, and path, including file name, to the DH file to be imported. For example, `http://www.example.com/dh_file`.

NOTE: The import fails if the file is on an HTTPS server that requires client certificate authentication for access.

Example

```
import ssl dhfile my-dhfile http://www.example.com/dh_file
```

rm ssl dhFile

Deletes the specified DH file.

Synopsys

```
rm ssl dhFile <name>
```

Arguments

name

Name of the DH file to delete.

Example

```
rm ssl dhfile my-dhfile
```

show ssl dhFile

Displays a list of all the imported DH file objects on the NetScaler ADC.

Synopsys

```
show ssl dhFile
```

Outputs

name

Name to assign to the imported DH file. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. The following requirement applies only to the NetScaler CLI: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my file" or 'my file').

src

URL specifying the protocol, host, and path, including file name, to the DH file to be imported. For example, `http://www.example.com/dh_file`.

NOTE: The import fails if the file is on an HTTPS server that requires client certificate authentication for access.

devno

count

stateflag

Example

```
show ssl dhfile
```

ssl dhParam

The following operations can be performed on "ssl dhParam":

create ssl dhParam

Generates a Diffie-Hellman (DH) key.

Synopsys

```
create ssl dhParam <dhFile> [<bits>] [-gen ( 2 | 5 )]
```

Arguments

dhFile

Name of and, optionally, path to the DH key file. /nsconfig/ssl/ is the default path.

Maximum value: 63

bits

Size, in bits, of the DH key being generated.

Minimum value: 512

Maximum value: 2048

gen

Random number required for generating the DH key. Required as part of the DH key generation algorithm.

Possible values: 2, 5

Default value: 2

Example

```
1) create ssl dhparam /nsconfig/ssl/dh1024.pem 1024 -gen 5
```

ssl dsaKey

The following operations can be performed on "ssl dsaKey":

create ssl dsaKey

Generates a DSA key.

Synopsys

```
create ssl dsaKey <keyFile> <bits> [-keyform ( DER | PEM )] [-des | -des3] {-password }
```

Arguments

keyFile

Name for and, optionally, path to the DSA key file. /nsconfig/ssl/ is the default path.

Maximum value: 63

bits

Size, in bits, of the DSA key.

Minimum value: 512

Maximum value: 2048

keyform

Format in which the DSA key file is stored on the appliance.

Possible values: DER, PEM

Default value: PEM

des

Encrypt the generated DSA key by using the DES algorithm. On the command line, you are prompted to enter the pass phrase (password) that will be used to encrypt the key.

des3

Encrypt the generated DSA key by using the Triple-DES algorithm. On the command line, you are prompted to enter the pass phrase (password) that will be used to encrypt the key.

password

Pass phrase to use for encryption if DES or DES3 option is selected.

Maximum value: 31

Example

```
create ssl dsakey /nsconfig/ssl/dsa1024.pem 1024
```

ssl dtlsProfile

The following operations can be performed on "ssl dtlsProfile":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add ssl dtlsProfile

Create a new DTLS profile on the NetScaler ADC.

Synopsys

```
add ssl dtlsProfile <name> [-pmtuDiscovery ( ENABLED | DISABLED )] [-maxRecordSize <positive_integer>] [-maxRetryTime <positive_integer>] [-helloVerifyRequest ( ENABLED | DISABLED )] [-terminateSession ( ENABLED | DISABLED )] [-maxPacketSize <positive_integer>]
```

Arguments

name

Name for the DTLS profile. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals sign (=), and hyphen (-) characters. Cannot be changed after the profile is created.

pmtuDiscovery

Source for the maximum record size value. If ENABLED, the value is taken from the PMTU table. If DISABLED, the value is taken from the profile.

Possible values: ENABLED, DISABLED

Default value: DISABLED

maxRecordSize

Maximum size of records that can be sent if PMTU is disabled.

Default value: 1459

Minimum value: 250

Maximum value: 1459

maxRetryTime

Wait for the specified time, in seconds, before resending the request.

Default value: 3

Minimum value: 0

helloVerifyRequest

Send a Hello Verify request to validate the client.

Possible values: ENABLED, DISABLED

Default value: DISABLED

terminateSession

Terminate the session if the message authentication code (MAC) of the client and server do not match.

Possible values: ENABLED, DISABLED

Default value: DISABLED

maxPacketSize

Maximum number of packets to reassemble. This value helps protect against a fragmented packet attack.

Default value: 120

Minimum value: 0

Maximum value: 86400

Example

```
add dtlsProfile dtls1 -helloVerifyRequest ENABLED -maxRetryTime 4
```

rm ssl dtlsProfile

Remove a DTLS profile on the Netscaler

Synopsys

```
rm ssl dtlsProfile <name>
```

Arguments

name

Name of the DTLS profile

Example

```
rm dtlsprofile <profile name>
```

set ssl dtlsProfile

Set/modify DTLS profile values

Synopsys

```
set ssl dtlsProfile <name> [-pmtuDiscovery ( ENABLED | DISABLED )] [-maxRecordSize <positive_integer>] [-maxRetryTime <positive_integer>] [-helloVerifyRequest ( ENABLED | DISABLED )] [-terminateSession ( ENABLED | DISABLED )] [-maxPacketSize <positive_integer>]
```

Arguments

name

Name for the DTLS profile. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals sign (=), and hyphen (-) characters. Cannot be changed after the profile is created.

pmtuDiscovery

Source for the maximum record size value. If ENABLED, the value is taken from the PMTU table. If DISABLED, the value is taken from the profile.

Possible values: ENABLED, DISABLED

Default value: DISABLED

maxRecordSize

Maximum size of records that can be sent if PMTU is disabled.

Default value: 1459

Minimum value: 250

Maximum value: 1459

maxRetryTime

Wait for the specified time, in seconds, before resending the request.

Default value: 3

Minimum value: 0

helloVerifyRequest

Send a Hello Verify request to validate the client.

Possible values: ENABLED, DISABLED

Default value: DISABLED

terminateSession

Terminate the session if the message authentication code (MAC) of the client and server do not match.

Possible values: ENABLED, DISABLED

Default value: DISABLED

maxPacketSize

Maximum number of packets to reassemble. This value helps protect against a fragmented packet attack.

Default value: 120

Minimum value: 0

Maximum value: 86400

Example

```
set dtlsprofile <profile name> -dropInvalReqs ON -markHttp09Inval ON
```

unset ssl dtlsProfile

Use this command to remove ssl dtlsProfile settings. Refer to the set ssl dtlsProfile command for meanings of the arguments.

Synopsis

```
unset ssl dtlsProfile <name> [-pmtuDiscovery] [-maxRecordSize] [-maxRetryTime] [-helloVerifyRequest] [-terminateSession] [-maxPacketSize]
```

show ssl dtlsProfile

Display all the configured DTLS profiles in the system. If a name is specified, then only that profile is shown.

Synopsis

```
show ssl dtlsProfile [<name>]
```

Arguments

name

Name of the DTLS profile.

Outputs

pmtuDiscovery

PMTU Discovery

maxRecordSize

Maximum record size

maxRetryTime

Maximum retry time

helloVerifyRequest

Hello Verify Request

terminateSession

Terminate Session

maxPacketSize

Maximum Packet Size

devno**count****stateflag**

Example

```
show dtls profile [profile name]
```

ssl fips

The following operations can be performed on "ssl fips":

set | **unset** | **reset** | **show** | **update**

set ssl fips

Initializes the Hardware Security Module (HSM) on the FIPS card and sets a new security officer password and user password. CAUTION: This command erases all data on the FIPS card. You are prompted before proceeding with the command execution. A restart is required before and after executing this command for the changes to apply. Save the configuration after executing this command and before restarting the appliance.

Synopsys

```
set ssl fips -initHSM Level-2 [-hsmLabel <string>]
```

Arguments

initHSM

FIPS initialization level. The appliance currently supports Level-2 (FIPS 140-2).

Possible values: Level-2

soPassword

Security officer password that will be in effect after you have configured the HSM.

oldSoPassword

Old password for the security officer.

userPassword

The Hardware Security Module's (HSM) User password.

hsmLabel

Label to identify the Hardware Security Module (HSM).

Example

```
1) set fips -initHSM Level-2 fipssol23 oldfipssol23 fipuser123 -hsmLabel FIPS-140-2 >This
```

unset ssl fips

Use this command to remove ssl fips settings. Refer to the set ssl fips command for meanings of the arguments.

Synopsys

```
unset ssl fips -hsmLabel
```

reset ssl fips

Resets the FIPS card to the default password for Security Officer and User accounts. This command can be used only if the FIPS card has been locked because of three or more unsuccessful login attempts.

Synopsys

```
reset ssl fips
```

Example

reset fips

show ssl fips

Displays the information on the FIPS card.

Synopsys

show ssl fips

Outputs

initHSM

The level of the FIPS initialization.

soPassword

Security officer password that will be in effect after you have configured the HSM.

userPassword

The Hardware Security Module's (HSM) User password.

oldSoPassword

Old password for the security officer.

eraseData

Erase data.

hsmLabel

FIPS card (HSM) label

serial

FIPS card serial number.

majorVersion

Firmware major version.

minorVersion

Firmware minor version.

FipsHwMajorVersion

FIPS card hardware major version.

FipsHwMinorVersion

FIPS card hardware minor version.

FipsHwVersionString

FIPS card hardware extended version string.

flashMemoryTotal

Total size of the flash memory on card.

flashMemoryFree

Total size of free flash memory.

sramTotal

Total size of the SRAM memory on card.

sramFree

Total size of free SRAM memory.

status

Status.

flag

Internal Flags.

serialNo

FIPS card serial number.

model

FIPS card model info.

state

FIPS card state.

firmwareReleaseDate

FIPS card firmware revision date.

coresMax

Maximum number of crypto cores present in the FIPS card.

coresEnabled

Number of crypto cores enabled in the FIPS card.

Example

An example of the output for `show ssl fips` command is as follows: `FIPS HSM Info: HSM L:`

update ssl fips

Updates the FIPS firmware. Note: Only compatible firmware version upgrade is allowed. For example, 4.6.0 to 4.6.1

Synopsys

```
update ssl fips -fipsFW 4.6.1
```

Arguments

fipsFW

FIPS firmware update.

Possible values: 4.6.1

Example

```
update ssl fips -fipsFW 4.6.1
```

ssl fipsKey

The following operations can be performed on "ssl fipsKey":

[create](#) | [rm](#) | [show](#) | [import](#) | [export](#)

create ssl fipsKey

Generates a FIPS key within the Hardware Security Module (HSM) of the FIPS card.

Synopsys

```
create ssl fipsKey <fipsKeyName> -modulus <positive_integer> [-exponent ( 3 | F4 )]
```

Arguments

fipsKeyName

Name for the FIPS key. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the FIPS key is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my fipskey" or 'my fipskey').

modulus

Modulus, in multiples of 64, of the FIPS key to be created.

Minimum value: 1024

Maximum value: 4096

exponent

Exponent value for the FIPS key to be created. Available values function as follows:

3=3 (hexadecimal)

F4=10001 (hexadecimal)

Possible values: 3, F4

Default value: 3

Example

```
create fipskey fips1 -modulus 1024 -exp f4
```

rm ssl fipsKey

Removes all the FIPS keys, or the specified FIPS key, from the appliance.

Synopsys

```
rm ssl fipsKey <fipsKeyName> ...
```

Arguments

fipsKeyName

Name of the FIPS key to remove.

Example

```
rm fipskey fips1
```

show ssl fipsKey

Displays information about all the FIPS keys configured on the appliance, or displays detailed information about the specified FIPS key.

Synopsys

```
show ssl fipsKey [<fipsKeyName>]
```

Arguments

fipsKeyName

Name of the FIPS key for which to show detailed information.

Outputs

modulus

The modulus of the key.

exponent

The exponent value for the key.

size

Size.

devno

count

stateflag

Example

1) An example of output of show ssl fipskey command is as follows: show fipskey 2 FIPS key

import ssl fipsKey

Imports a FIPS key into the Hardware Security Module (HSM) of the FIPS card. Can import an existing FIPS key, or can import, as a FIPS key, an external private key, such as a key that was created on an Apache or IIS external Web server.

Synopsys

```
import ssl fipsKey <fipsKeyName> -key <string> [-inform <inform>] [-wrapKeyName <string>] [-iv <string>] [-exponent ( 3 | F4 )]
```

Arguments

fipsKeyName

Name for the FIPS key to be imported. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the FIPS key is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my fipskey" or 'my fipskey').

key

Name of and, optionally, path to the key file to be imported.

/nsconfig/ssl/ is the default path.

inform

Input format of the key file. Available formats are:

SIM - Secure Information Management; select when importing a FIPS key. If the external FIPS key is encrypted, first decrypt it, and then import it.

PEM - Privacy Enhanced Mail; select when importing a non-FIPS key.

Possible values: SIM, DER, PEM

Default value: SIM

wrapKeyName

Name of the wrap key to use for importing the key. Required for importing a non-FIPS key.

iv

Initialization Vector (IV) to use for importing the key. Required for importing a non-FIPS key.

exponent

Exponent value for the FIPS key to be created. Available values function as follows:

3=3 (hexadecimal)

F4=10001 (hexadecimal)

Possible values: 3, F4

Default value: 3

Example

```
1) import fipskey fips1 -key /nsconfig/ssl/fipskey.sim
```

 The above example imports a FIPS key

export ssl fipsKey

Exports a FIPS key from one appliance to another or backs up a FIPS key in a secure manner. The exported key is secured by using a strong asymmetric key encryption method.

Synopsis

```
export ssl fipsKey <fipsKeyName> -key <string>
```

Arguments

fipsKeyName

Name of the FIPS key to export.

key

Name of and, optionally, path to the exported key file.

/nsconfig/ssl/ is the default path.

Example

```
export fipskey fips1 -key /nsconfig/ssl/fips1.key
```

ssl fipsSIMSource

The following operations can be performed on "ssl fipsSIMSource":

[enable](#) | [init](#)

enable ssl fipsSIMSource

Enable the source FIPS appliance to participate in a secure exchange of keys with the target (secondary) FIPS appliance.

Synopsis

```
enable ssl fipsSIMSource <targetSecret> <sourceSecret>
```

Arguments

targetSecret

Name of and, optionally, path to the target FIPS appliance's secret data. /nsconfig/ssl/ is the default path.

sourceSecret

Name for and, optionally, path to the source FIPS appliance's secret data. /nsconfig/ssl/ is the default path.

Example

```
enable fipsSIMsource /nsconfig/ssl/target.secret /nsconfig/ssl/source.secret
```

init ssl fipsSIMSource

Initialize the source FIPS appliance for participating in a secure exchange of keys with the target (secondary) FIPS appliance.

Synopsis

```
init ssl fipsSIMSource <certFile>
```

Arguments

certFile

Name for and, optionally, path to the source FIPS appliance's certificate file. /nsconfig/ssl/ is the default path.

Example

```
init fipsSIMsource /nsconfig/ssl/source.cert
```

ssl fipsSIMTarget

The following operations can be performed on "ssl fipsSIMTarget":

[enable](#) | [init](#)

enable ssl fipsSIMTarget

Enables secure transfer of FIPS keys in a high availability setup from the primary appliance to the secondary appliance.

Synopsis

```
enable ssl fipsSIMTarget <keyVector> <sourceSecret>
```

Arguments

keyVector

Name of and, optionally, path to the target FIPS appliance's key vector. /nsconfig/ssl/ is the default path.

sourceSecret

Name of and, optionally, path to the source FIPS appliance's secret data. /nsconfig/ssl/ is the default path.

Example

```
enable fipsSIMtarget /nsconfig/ssl/target.key /nsconfig/ssl/source.secret
```

init ssl fipsSIMTarget

Initialize the target (secondary) FIPS appliance for participating in a secure exchange of keys with the primary FIPS appliance.

Synopsis

```
init ssl fipsSIMTarget <certFile> <keyVector> <targetSecret>
```

Arguments

certFile

Name of and, optionally, path to the source FIPS appliance's certificate file. /nsconfig/ssl/ is the default path.

keyVector

Name for and, optionally, path to the target FIPS appliance's key vector. /nsconfig/ssl/ is the default path.

targetSecret

Name for and, optionally, path to the target FIPS appliance's secret data. The default input path for the secret data is /nsconfig/ssl/.

Example

```
init fipsSIMtarget /nsconfig/ssl/source.cert /nsconfig/ssl/target.key /nsconfig/ssl/targetSecret
```

ssl global

The following operations can be performed on "ssl global":

[bind](#) | [unbind](#) | [show](#)

bind ssl global

Binds an SSL policy globally.

Synopsys

```
bind ssl global [-policyName <string>] [-priority <positive_integer>] [-gotoPriorityExpression <expression>] [-type <type>] [-invoke (<labelType> <labelName>)]
```

Arguments

policyName

Name of the SSL policy.

priority

Integer specifying the policy's priority. The lower the number, the higher the priority.

Minimum value: 1

Maximum value: 2147483647

gotoPriorityExpression

Expression or other value specifying the next policy to be evaluated if the current policy evaluates to TRUE. Specify one of the following values:

- * NEXT - Evaluate the policy with the next higher priority number.
- * END - End policy evaluation.
- * USE_INVOCATION_RESULT - Applicable if this policy invokes another policy label. If the final goto in the invoked policy label has a value of END, the evaluation stops. If the final goto is anything other than END, the current policy label performs a NEXT.
- * A default syntax expression that evaluates to a number.

If you specify an expression, the number to which it evaluates determines the next policy to evaluate, as follows:

- * If the expression evaluates to a higher numbered priority, the policy with that priority is evaluated next.
- * If the expression evaluates to the priority of the current policy, the policy with the next higher numbered priority is evaluated next.
- * If the expression evaluates to a number that is larger than the largest numbered priority, policy evaluation ends.

An UNDEF event is triggered if:

- * The expression is invalid.
- * The expression evaluates to a priority number that is numerically lower than the current policy's priority.
- * The expression evaluates to a priority number that is between the current policy's priority number (say, 30) and the highest priority number (say, 100), but does not match any configured priority number (for example, the expression evaluates to the number 85). This example assumes that the priority number increments by 10 for every successive policy, and therefore a priority number of 85 does not exist in the policy label.

Default value: "END"

type

Global bind point to which to bind the default syntax policy. If CONTROL_OVERRIDE or DATA_OVERRIDE is selected, the global control or data policy overrides the control or data policy bound to the virtual server or service.

Possible values: CONTROL_OVERRIDE, CONTROL_DEFAULT, DATA_OVERRIDE, DATA_DEFAULT

invoke

Invoke policies bound to a virtual server, service, or policy label. After the invoked policies are evaluated, the flow returns to the policy with the next priority.

labelType

Type of policy label to invoke. Specify virtual server for a policy label associated with a virtual server, or policy label for a user-defined policy label.

Possible values: vserver, service, policylabel

labelName

Name of the virtual server or user-defined policy label to invoke if the policy evaluates to TRUE.

Example

```
bind ssl global -policyName certInsert_pol -priority 100
```

unbind ssl global

Unbinds a globally bound SSL policy.

Synopsys

```
unbind ssl global [-policyName <string> [-type <type>] [-priority <positive_integer>]]
```

Arguments

policyName

Name of the SSL policy to unbind.

type

Global bind point from which the policy is to be unbound.

Possible values: CONTROL_OVERRIDE, CONTROL_DEFAULT, DATA_OVERRIDE, DATA_DEFAULT

priority

Priority of the NOPOLICY (built-in policy) to be unbound. Not required if you are unbinding a user-defined policy.

Minimum value: 1

Maximum value: 2147483647

Example

```
unbind ssl global -policyName certInsert_pol
```

show ssl global

Displays globally bound SSL policies.

Synopsys

show ssl global [-type <type>]

Arguments

type

Global bind point to which the policy is bound.

Possible values: CONTROL_OVERRIDE, CONTROL_DEFAULT, DATA_OVERRIDE, DATA_DEFAULT

Outputs

stateflag

policyName

The name for the SSL policy.

priority

The priority of the policy binding.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE

invoke

Invoke flag. This attribute is relevant only for ADVANCED policies

labelType

Type of policy label invocation.

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

numpol

The number of policies bound to the bindpoint.

devno

count

Example

```
show ssl global          1 Globally Active SSL Policy: 1)      Name: certInsert_pol
```

ssl keyFile

The following operations can be performed on "ssl keyFile":

[import](#) | [rm](#) | [show](#)

import ssl keyFile

Imports a key file to the NetScaler appliance, assigns it a name, and stores it in the /nsconfig/ssl/keyfilefolder. The folder is created if it does not exist.

Synopsys

```
import ssl keyFile <name> <src>
```

Arguments

name

Name to assign to the imported key file. Must begin with an ASCII alphanumeric or underscore(_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. The following requirement applies only to the NetScaler CLI: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my file" or 'my file').

src

URL specifying the protocol, host, and path, including file name, to the key file to be imported. For example, `http://www.example.com/key_file`.

NOTE: The import fails if the object to be imported is on an HTTPS server that requires client certificate authentication for access.

Example

```
import ssl keyfile my-keyfile http://www.example.com/key_file
```

rm ssl keyFile

Deletes the specified key file.

Synopsys

```
rm ssl keyFile <name>
```

Arguments

name

Name of the key file to be delete.

Example

```
rm ssl keyfile <name>
```

show ssl keyFile

Displays lists of all the imported key file objects on the NetScaler ADC.

Synopsys

```
show ssl keyFile
```

Outputs

name

Name to assign to the imported key file. Must begin with an ASCII alphanumeric or underscore(_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. The following requirement applies only to the NetScaler CLI: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my file" or 'my file').

src

URL specifying the protocol, host, and path, including file name, to the key file to be imported. For example, `http://www.example.com/key_file`.

NOTE: The import fails if the object to be imported is on an HTTPS server that requires client certificate authentication for access.

devno

count

stateflag

Example

```
show ssl keyfile
```


ssl ocsponder

The following operations can be performed on "ssl ocsponder":

add | **rm** | **set** | **unset** | **show**

add ssl ocsponder

Adds an OCSponder responder. An OCSponder responder identifies the OCSponder server that validates a certificate. NetScaler appliances support OCSponder as defined in RFC 2560.

Synopsis

```
add ssl ocsponder <name> -url <URL> [-cache ( ENABLED | DISABLED ) [-cacheTimeout <positive_integer>]]  
[-batchingDepth <positive_integer>] [-batchingDelay <positive_integer>] [-resptimeout <positive_integer>] [-  
responderCert <string> | -trustResponder] [-producedAtTimeSkew <positive_integer>] [-signingCert <string>] [-  
useNonce ( YES | NO )] [-insertClientCert ( YES | NO )]
```

Arguments

name

Name for the OCSponder responder. Cannot begin with a hash (#) or space character and must contain only ASCII alphanumeric, underscore (_), hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the responder is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my responder" or 'my responder').

url

URL of the OCSponder responder.

cache

Enable caching of responses. Caching of responses received from the OCSponder responder enables faster responses to the clients and reduces the load on the OCSponder responder.

Possible values: ENABLED, DISABLED

cacheTimeout

Timeout for caching the OCSponder response. After the timeout, the NetScaler sends a fresh request to the OCSponder responder for the certificate status. If a timeout is not specified, the timeout provided in the OCSponder response applies.

Default value: 1

Minimum value: 1

Maximum value: 1440

batchingDepth

Number of client certificates to batch together into one OCSponder request. Batching avoids overloading the OCSponder responder. A value of 1 signifies that each request is queried independently. For a value greater than 1, specify a timeout (batching delay) to avoid inordinately delaying the processing of a single certificate.

Minimum value: 1

Maximum value: 8

batchingDelay

Maximum time, in milliseconds, to wait to accumulate OCSponder requests to batch. Does not apply if the Batching Depth is 1.

Minimum value: 0

Maximum value: 10000

resptimeout

Time, in milliseconds, to wait for an OCSP response. When this time elapses, an error message appears or the transaction is forwarded, depending on the settings on the virtual server. Includes Batching Delay time.

Minimum value: 0

Maximum value: 120000

responderCert

trustResponder

A certificate to use to validate OCSP responses. Alternatively, if -trustResponder is specified, no verification will be done on the response. If both are omitted, only the response times (producedAt, lastUpdate, nextUpdate) will be verified.

producedAtTimeSkew

Time, in seconds, for which the NetScaler waits before considering the response as invalid. The response is considered invalid if the Produced At time stamp in the OCSP response exceeds or precedes the current NetScaler clock time by the amount of time specified.

Default value: 300

Minimum value: 0

Maximum value: 86400

signingCert

Certificate-key pair that is used to sign OCSP requests. If this parameter is not set, the requests are not signed.

useNonce

Enable the OCSP nonce extension, which is designed to prevent replay attacks.

Possible values: YES, NO

insertClientCert

Include the complete client certificate in the OCSP request.

Possible values: YES, NO

Example

```
1) add ssl ocspResponder -url http://ocsp.example.com -producedAtTimeSkew 0
```

 The above command

rm ssl ocspResponder

Removes the specified OCSP responder from the appliance.

Synopsis

```
rm ssl ocspResponder <name> ...
```

Arguments

name

Name of the OCSP responder to remove. The OCSP responder is removed only if it is not referenced by any other object.

Example

1) `rm ssl ocspResponder 01` The above command removes the OCSP responder 01 from the system

set ssl ocspResponder

Modifies the parameters of an OCSP responder.

Synopsis

```
set ssl ocspResponder <name> [-url <URL>] [-cache ( ENABLED | DISABLED )] [-cacheTimeout <positive_integer>]
[-batchingDepth <positive_integer>] [-batchingDelay <positive_integer>] [-resptimeout <positive_integer>] [-
responderCert <string> | -trustResponder] [-producedAtTimeSkew <positive_integer>] [-signingCert <string>] [-
useNonce ( YES | NO )] [-insertClientCert ( YES | NO )]
```

Arguments

name

Name of the OCSP responder to modify.

url

URL of the OCSP responder.

cache

Enable caching of responses. Caching of responses received from the OCSP responder enables faster responses to the clients and reduces the load on the OCSP responder.

Possible values: ENABLED, DISABLED

cacheTimeout

Timeout for caching the OCSP response. After the timeout, the NetScaler sends a fresh request to the OCSP responder for the certificate status. If a timeout is not specified, the timeout provided in the OCSP response applies.

Default value: 1

Minimum value: 1

Maximum value: 1440

batchingDepth

Number of client certificates to batch together into one OCSP request. Batching avoids overloading the OCSP responder. A value of 1 signifies that each request is queried independently. For a value greater than 1, specify a timeout (batching delay) to avoid inordinately delaying the processing of a single certificate.

Minimum value: 1

Maximum value: 8

batchingDelay

Maximum time, in milliseconds, to wait to accumulate OCSP requests to batch. Does not apply if the Batching Depth is 1.

Minimum value: 0

Maximum value: 10000

resptimeout

Time, in milliseconds, to wait for an OCSP response. When this time elapses, an error message appears or the transaction is forwarded, depending on the settings on the virtual server. Includes Batching Delay time.

Minimum value: 0

Maximum value: 120000

responderCert

trustResponder

A certificate to use to validate OCSP responses. Alternatively, if -trustResponder is specified, no verification will be done on the response. If both are omitted, only the response times (producedAt, lastUpdate, nextUpdate) will be verified.

producedAtTimeSkew

Time, in seconds, for which the NetScaler waits before considering the response as invalid. The response is considered invalid if the Produced At time stamp in the OCSP response exceeds or precedes the current NetScaler clock time by the amount of time specified.

Default value: 300

Minimum value: 0

Maximum value: 86400

signingCert

Certificate-key pair that is used to sign OCSP requests. If this parameter is not set, the requests are not signed.

useNonce

Enable the OCSP nonce extension, which is designed to prevent replay attacks.

Possible values: YES, NO

insertClientCert

Include the complete client certificate in the OCSP request.

Possible values: YES, NO

Example

```
1) add ssl ocsponder -url http://ocsp.example.com -producedAtTimeSkew 0 The above com
```

unset ssl ocsponder

Removes the attributes of an OCSP responder. Attributes for which a default value is available revert to their default values. Refer to the set ssl ocsponder command for descriptions of the arguments..Refer to the set ssl ocsponder command for meanings of the arguments.

Synopsis

```
unset ssl ocsponder <name> [-trustResponder] [-insertClientCert ( YES | NO )] [-cache] [-cacheTimeout] [-batchingDepth] [-batchingDelay] [-resptimeout] [-responderCert] [-producedAtTimeSkew] [-signingCert] [-useNonce]
```

show ssl ocsponder

Displays information about all the OCSP responders configured on the appliance, or displays detailed information about the specified OCSP responder.

Synopsis

```
show ssl ocsponder [<name>]
```

Arguments

name

Name of the OCSP responder for which to show detailed information.

Outputs

url

URL of the OCSP responder.

useAIA

Only use the URL present in the certificate.

cache

Enable caching of responses. Caching of responses received from the OCSP responder enables faster responses to the clients and reduces the load on the OCSP responder.

cacheTimeout

Timeout for caching the OCSP response. After the timeout, the NetScaler sends a fresh request to the OCSP responder for the certificate status. If a timeout is not specified, the timeout provided in the OCSP response applies.

batchingDepth

Number of client certificates to batch together into one OCSP request. Batching avoids overloading the OCSP responder. A value of 1 signifies that each request is queried independently. For a value greater than 1, specify a timeout (batching delay) to avoid inordinately delaying the processing of a single certificate.

batchingDelay

Maximum time, in milliseconds, to wait to accumulate OCSP requests to batch. Does not apply if the Batching Depth is 1.

resptimeout

Maximum time, in mS, to wait for an OCSP response before giving up. Defaults to 2000 mS. If this is set to 0, NetScaler will wait for an indefinite amount of time.

producedAtTimeSkew

Time, in seconds, for which the NetScaler waits before considering the response as invalid. The response is considered invalid if the Produced At time stamp in the OCSP response exceeds or precedes the current NetScaler clock time by the amount of time specified.

responderCert**trustResponder**

A certificate to use to validate OCSP responses. Alternatively, if -trustResponder is specified, no verification will be done on the response. If both are omitted, only the response times (producedAt, lastUpdate, nextUpdate) will be verified.

signingCert

Certificate-key pair that is used to sign OCSP requests. If this parameter is not set, the requests are not signed.

useNonce

Add a nonce to the OCSP request. Protects against replay attacks.

dns

Was DNS resolution successful for a domain-based OCSP responder

insertClientCert

Include the complete client certificate in the OCSP request.

IPAddress

The IPv6 address of the ocsp responder.

devno

count

stateflag

ssl parameter

The following operations can be performed on "ssl parameter":

[set](#) | [unset](#) | [show](#)

set ssl parameter

Synopsys

```
set ssl parameter [-quantumSize <quantumSize>] [-crlMemorySizeMB <positive_integer>] [-strictCAChecks ( YES | NO )] [-sslTriggerTimeout <positive_integer>] [-sendCloseNotify ( YES | NO )] [-encryptTriggerPktCount <positive_integer>] [-denySSLReneg <denySSLReneg>] [-insertionEncoding ( Unicode | UTF-8 )] [-ocspCacheSize <positive_integer>] [-pushFlag <positive_integer>] [-dropReqWithNoHostHeader ( YES | NO )] [-pushEncTriggerTimeout <positive_integer>] [-cryptodevDisableLimit <positive_integer>] [-undefActionCode <string>] [-undefActionCode <string>]
```

Arguments

quantumSize

Amount of data to collect before the data is pushed to the crypto hardware for encryption. For large downloads, a larger quantum size better utilizes the crypto resources.

Possible values: 4096, 8192, 16384

Default value: 8192

crlMemorySizeMB

Maximum memory size to use for certificate revocation lists (CRLs). This parameter reserves memory for a CRL but sets a limit to the maximum memory that the CRLs loaded on the appliance can consume.

Default value: 256

Minimum value: 10

Maximum value: 1024

strictCAChecks

Enable strict CA certificate checks on the appliance.

Possible values: YES, NO

Default value: NO

sslTriggerTimeout

Time, in milliseconds, after which encryption is triggered for transactions that are not tracked on the NetScaler appliance because their length is not known. There can be a delay of up to 10ms from the specified timeout value before the packet is pushed into the queue.

Default value: 100

Minimum value: 1

Maximum value: 200

sendCloseNotify

Send an SSL Close-Notify message to the client at the end of a transaction.

Possible values: YES, NO

Default value: YES

encryptTriggerPktCount

Maximum number of queued packets after which encryption is triggered. Use this setting for SSL transactions that send small packets from server to NetScaler.

Default value: 45

Minimum value: 10

Maximum value: 50

denySSLReneg

Deny renegotiation in specified circumstances. Available settings function as follows:

* NO - Allow SSL renegotiation.

* FRONTEND_CLIENT - Deny secure and nonsecure SSL renegotiation initiated by the client.

* FRONTEND_CLIENTSERVER - Deny secure and nonsecure SSL renegotiation initiated by the client or the NetScaler during policy-based client authentication.

* ALL - Deny all secure and nonsecure SSL renegotiation.

* NONSECURE - Deny nonsecure SSL renegotiation. Allows only clients that support RFC 5746.

Possible values: NO, FRONTEND_CLIENT, FRONTEND_CLIENTSERVER, ALL, NONSECURE

Default value: ALL

insertionEncoding

Encoding method used to insert the subject or issuer's name in HTTP requests to servers.

Possible values: Unicode, UTF-8

Default value: Unicode

ocspCacheSize

Size, per packet engine, in megabytes, of the OCSP cache. A maximum of 10% of the packet engine memory can be assigned. Because the maximum allowed packet engine memory is 4GB, the maximum value that can be assigned to the OCSP cache is approximately 410 MB.

Default value: 10

Minimum value: 0

Maximum value: 512

pushFlag

Insert PUSH flag into decrypted, encrypted, or all records. If the PUSH flag is set to a value other than 0, the buffered records are forwarded on the basis of the value of the PUSH flag. Available settings function as follows:

0 - Auto (PUSH flag is not set.)

1 - Insert PUSH flag into every decrypted record.

2 - Insert PUSH flag into every encrypted record.

3 - Insert PUSH flag into every decrypted and encrypted record.

Minimum value: 0

Maximum value: 3

dropReqWithNoHostHeader

Host header check for SNI enabled sessions. If this check is enabled and the HTTP request does not contain the host header for SNI enabled sessions, the request is dropped.

Possible values: YES, NO

Default value: NO

pushEncTriggerTimeout

PUSH encryption trigger timeout value. The timeout value is applied only if you set the Push Encryption Trigger parameter to Timer in the SSL virtual server settings.

Default value: 1

Minimum value: 1

Maximum value: 200

cryptodevDisableLimit

Disabled Crypto Device Limit reboots the system once reached. A value of zero(0) implies no reboot.

Default value: 0

Minimum value: 0

undefActionControl

Name of the undefined built-in control action: CLIENTAUTH, NOCLIENTAUTH, NOOP, RESET, or DROP.

Default value: "CLIENTAUTH"

undefActionData

Name of the undefined built-in data action: NOOP, RESET or DROP.

Default value: "NOOP"

unset ssl parameter

Use this command to remove ssl parameter settings. Refer to the set ssl parameter command for meanings of the arguments.

Synopsis

```
unset ssl parameter [-quantumSize] [-crlMemorySizeMB] [-strictCAGhecks] [-sslTriggerTimeout] [-sendCloseNotify] [-encryptTriggerPktCount] [-denySSLReneg] [-insertionEncoding] [-ocspCacheSize] [-pushFlag] [-dropReqWithNoHostHeader] [-pushEncTriggerTimeout] [-cryptodevDisableLimit] [-undefActionControl] [-undefActionData]
```

show ssl parameter

Displays information about advanced SSL parameters.

Synopsis

```
show ssl parameter
```

Outputs

quantumSize

Amount of data to collect before the data is pushed to the crypto hardware for encryption. For large downloads, a larger quantum size better utilizes the crypto resources.

crlMemorySizeMB

Maximum memory size to use for certificate revocation lists (CRLs). This parameter reserves memory for a CRL but sets a limit to the maximum memory that the CRLs loaded on the appliance can consume.

strictCAChecks

Memory size to use for CRLs

sslTriggerTimeout

Encryption trigger timer. Set the encryption trigger timeout for transactions, which are not trackable by Netscaler. NetScaler will use this setting to accumulate data received from the server for the configured time period before pushing it to the crypto hardware for encryption.

sendCloseNotify

Send an SSL Close-Notify message to the client at the end of a transaction.

encryptTriggerPktCount

Maximum number of queued packets after which encryption is triggered. Use this setting for SSL transactions that send small packets from server to NetScaler.

denySSLReneg

SSL Renegotiation setting

insertionEncoding

Encoding method used to insert the subject or issuer's name in HTTP requests to servers.

ocspCacheSize

Size, per packet engine, in megabytes of the OCSP cache

pushFlag

Insert PUSH flag into decrypted, encrypted, or all records. If the PUSH flag is set to a value other than 0, the buffered records are forwarded on the basis of the value of the PUSH flag. Available settings function as follows:

0 - Auto (PUSH flag is not set.)

1 - Insert PUSH flag into every decrypted record.

2 - Insert PUSH flag into every encrypted record.

3 - Insert PUSH flag into every decrypted and encrypted record.

dropReqWithNoHostHeader

Host header check for SNI enabled sessions. If this check is enabled and the HTTP request does not contain the host header for SNI enabled sessions, the request is dropped.

pushEncTriggerTimeout

PUSH encryption trigger timeout value. The timeout value is applied only if you set the Push Encryption Trigger parameter to Timer in the SSL virtual server settings.

cryptodevDisableLimit

Disabled Crypto Device Limit reboots the system once reached. A value of zero(0) implies no reboot

undefActionControl

Global undef action for SSL control policies

undefActionData

Global undef action for SSL data policies

ssl pkcs12

The following operations can be performed on "ssl pkcs12":

convert ssl pkcs12

Converts the end-user certificate from PEM encoding format to PKCS#12 format. This certificate can then be distributed and installed in browsers as client certificates.

Synopsys

```
convert ssl pkcs12 <outfile> [-import [-pkcs12File <input_filename>] [-des | -des3] ] [-export [-certFile  
<input_filename>] [-keyFile <input_filename>]] {-password } {-PEMPassPhrase }
```

Arguments

outfile

Name for and, optionally, path to, the output file that contains the certificate and the private key after converting from PKCS#12 to PEM format. /nsconfig/ssl/ is the default path.

If importing, the certificate-key pair is stored in PEM format. If exporting, the certificate-key pair is stored in PKCS#12 format.

Maximum value: 63

import

Convert the certificate and private-key from PKCS#12 format to PEM format.

pkcs12File

Name for and, optionally, path to, the PKCS#12 file. If importing, specify the input file name that contains the certificate and the private key in PKCS#12 format. If exporting, specify the output file name that contains the certificate and the private key after converting from PEM to

PKCS#12 format. /nsconfig/ssl/ is the default path.

During the import operation, if the key is encrypted, you are prompted to enter the pass phrase used for encrypting the key.

Maximum value: 63

des

Encrypt the private key by using the DES algorithm in CBC mode during the import operation. On the command line, you are prompted to enter the pass phrase.

des3

Encrypt the private key by using the Triple-DES algorithm in EDE CBC mode (168-bit key) during the import operation. On the command line, you are prompted to enter the pass phrase.

export

Convert the certificate and private key from PEM format to PKCS#12 format. On the command line, you are prompted to enter the pass phrase.

certFile

Certificate file to be converted from PEM to PKCS#12 format.

Maximum value: 63

keyFile

Name of the private key file to be converted from PEM to PKCS#12 format. If the key file is encrypted, you are prompted to enter the pass phrase used for encrypting the key.

Maximum value: 63

password

PEMPassPhrase

Example

```
1) convert ssl pkcs12 /nsconfig/ssl/client_certkey.pl2 -export -cert /nsconfig/ssl/cli
```

ssl pkcs8

The following operations can be performed on "ssl pkcs8":

convert ssl pkcs8

Convert a PEM or DER format key file to PKCS#8 format before importing it into the FIPS appliance.

Synopsys

```
convert ssl pkcs8 <pkcs8File> <keyFile> [-keyform ( DER | PEM )] {-password }
```

Arguments

pkcs8File

Name for and, optionally, path to, the output file where the PKCS#8 format key file is stored. /nsconfig/ssl/ is the default path.

Maximum value: 63

keyFile

Name of and, optionally, path to the input key file to be converted from PEM or DER format to PKCS#8 format. /nsconfig/ssl/ is the default path.

Maximum value: 63

keyform

Format in which the key file is stored on the appliance.

Possible values: DER, PEM

Default value: PEM

password

Password to assign to the file if the key is encrypted. Applies only for PEM format files.

Maximum value: 31

Example

```
convert ssl pkcs8 /nsconfig/ssl/key.pk8 /nsconfig/ssl/key.pem
```

ssl policy

The following operations can be performed on "ssl policy":

add | **rm** | **set** | **unset** | **show**

add ssl policy

Adds an SSL policy. An SSL policy evaluates incoming traffic and applies a predefined action to requests that match a rule (expression). You have to configure the actions before creating the policies, so that you can specify an action when you create a policy.

Synopsys

add ssl policy <name> -rule <expression> [-action <string>] [-undefAction <string>] [-comment <string>]

Arguments

name

Name for the new SSL policy. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my policy" or 'my policy').

rule

Expression, against which traffic is evaluated. Written in the classic or default syntax.

Note:

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

(Classic expressions are not supported in the cluster build.)

The following requirements apply only to the NetScaler CLI:

- * If the expression includes one or more spaces, enclose the entire expression in double quotation marks.
- * If the expression itself includes double quotation marks, escape the quotations by using the \ character.
- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Name of the built-in or user-defined action to perform on the request. Available built-in actions are NOOP, RESET, DROP, CLIENTAUTH, and NOCLIENTAUTH.

undefAction

Name of the action to be performed when the result of rule evaluation is undefined. Possible values for control policies: CLIENTAUTH, NOCLIENTAUTH, NOOP, RESET, DROP. Possible values for data policies: NOOP, RESET or DROP.

comment

Any comments associated with this policy.

Example

```
add ssl action certInsert_act -clientCert ENABLED -certHeader CERT add ssl policy cert:
```

rm ssl policy

Removes an SSL policy.

Synopsis

```
rm ssl policy <name>
```

Arguments

name

Name of the SSL policy to be removed.

Example

```
rm ssl policy certInsert_pol
```

set ssl policy

Modifies the parameters of an SSL default syntax policy.

Synopsis

```
set ssl policy <name> [-rule <expression>] [-action <string>] [-undefAction <string>] [-comment <string>]
```

Arguments

name

Name of the SSL policy to modify.

rule

Expression, against which traffic is evaluated. Written in the classic or default syntax.

Note:

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

(Classic expressions are not supported in the cluster build.)

The following requirements apply only to the NetScaler CLI:

- * If the expression includes one or more spaces, enclose the entire expression in double quotation marks.
- * If the expression itself includes double quotation marks, escape the quotations by using the \ character.
- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Name of the built-in or user-defined action to perform on the request. Available built-in actions are NOOP, RESET, DROP, CLIENTAUTH, and NOCLIENTAUTH.

undefAction

Name of the action to be performed when the result of rule evaluation is undefined. Possible values for control policies: CLIENTAUTH, NOCLIENTAUTH, NOOP, RESET, DROP. Possible values for data policies: NOOP, RESET or DROP.

comment

Any comments associated with this policy.

Example

```
set ssl policy poll -rule "HTTP.REQ.HEADER(\\\\"header\\\\" ).CONTAINS(\\\\"qh2\\\\" )" "
```

unset ssl policy

Removes the attributes of an SSL default syntax policy. Attributes for which a default value is available revert to their default values. Refer to the set ssl policy command for a description of the parameters..Refer to the set ssl policy command for meanings of the arguments.

Synopsys

```
unset ssl policy <name> [-undefAction] [-comment]
```

Example

```
unset ssl policy poll -undefAction
```

show ssl policy

Displays information about all the SSL policies configured on the appliance, or displays detailed information about the specified SSL policy.

Synopsys

```
show ssl policy [<name>]
```

Arguments

name

Name of the SSL policy for which to display detailed information.

Outputs

stateflag

rule

The expression that sets the condition for application of the SSL policy.

action

The name of the action to be performed on the request.

undefAction

Undef Action associated with the policy.

hits

Number of hits for this policy.

piHits

Number of hits.

undefHits

Number of Undef hits.

activePolicy

boundTo

The entity name to which policy is bound

priority

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

labelType

Type of policy label invocation.

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

description

Description of the policy

comment

Any comments associated with this policy.

bindPolicyType

vserverType

policyType

peFlags

devno

count

Example

```
show ssl policy 1 SSL policy: 1)      Name: certInsert_pol      Rule: URL == /*      Act:
```

ssl policylabel

The following operations can be performed on "ssl policylabel":

add | **rm** | **bind** | **unbind** | **show**

add ssl policylabel

Creates an SSL policy label. An SSL policy label can be a control label or a data label.

Synopsys

```
add ssl policylabel <labelName> -type ( CONTROL | DATA )
```

Arguments

labelName

Name for the SSL policy label. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the policy label is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my label" or 'my label').

type

Type of policies that the policy label can contain.

Possible values: CONTROL, DATA

Example

```
add ssl policylabel ssl_pol_label -type REQ
```

rm ssl policylabel

Removes an SSL policy label.

Synopsys

```
rm ssl policylabel <labelName>
```

Arguments

labelName

Name of the SSL policy label to remove.

Example

```
rm ssl policylabel ssl_pol_label
```

bind ssl policylabel

Binds an SSL policy to an SSL policy label and specifies the order in which the policies in the label are to be evaluated.

Synopsys

```
bind ssl policylabel <labelName> <policyName> <priority> [<gotoPriorityExpression>] [-invoke (<labelType>
<labelName>)]
```

Arguments

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

policyName

Name of the SSL policy to bind to the policy label.

priority

Integer specifying the policy's priority within the label. The lower the priority number, the higher the policy's priority. Policies are evaluated in order of priority, but the order can be modified by a goto priority expression.

Minimum value: 1

Maximum value: 2147483647

gotoPriorityExpression

Expression or other value specifying the next policy to be evaluated if the current policy evaluates to TRUE. Specify one of the following values:

- * NEXT - Evaluate the policy with the next higher priority number.
- * END - End policy evaluation.
- * USE_INVOCATION_RESULT - Applicable if this policy invokes another policy label. If the final goto in the invoked policy label has a value of END, the evaluation stops. If the final goto is anything other than END, the current policy label performs a NEXT.
- * A default syntax or classic expression that evaluates to a number.

If you specify an expression, the number to which it evaluates determines the next policy to evaluate, as follows:

- * If the expression evaluates to a higher numbered priority, the policy with that priority is evaluated next.
- * If the expression evaluates to the priority of the current policy, the policy with the next higher numbered priority is evaluated next.
- * If the expression evaluates to a number that is larger than the largest numbered priority, policy evaluation ends.

An UNDEF event is triggered if:

- * The expression is invalid.
- * The expression evaluates to a priority number that is numerically lower than the current policy's priority.
- * The expression evaluates to a priority number that is between the current policy's priority number (say, 30) and the highest priority number (say, 100), but does not match any configured priority number (for example, the expression evaluates to the number 85). This example assumes that the priority number increments by 10 for every successive policy, and therefore a priority number of 85 does not exist in the policy label.

Default value: "END"

invoke

Invoke policies bound to a policy label. After the invoked policies are evaluated, the flow returns to the policy with the next priority.

labelType

Type of policy label invocation.

Possible values: vserver, service, policylabel

Example

```
bind ssl policylabel ssl_pol_label -policyName ssl_pol -priority 1
```

unbind ssl policylabel

Unbinds an SSL policy from an SSL policy label.

Synopsys

```
unbind ssl policylabel <labelName> <policyName> [-priority <positive_integer>]
```

Arguments

labelName

Name of the SSL policy label from which to unbind policies.

policyName

Name of the SSL policy to unbind.

priority

Priority of the NOPOLICY to be unbound.

Minimum value: 1

Maximum value: 2147483647

Example

```
unbind ssl policylabel ssl_pol_label ssl_pol
```

show ssl policylabel

Displays information about all the SSL policy labels, or displays detailed information about the specified policy label.

Synopsys

```
show ssl policylabel [<labelName>]
```

Arguments

labelName

Name of the SSL policy label for which to show detailed information.

Outputs

stateflag

type

Type of policies that the policy label can contain.

numpol

Number of policies bound to label.

hits

Number of times policy label was invoked.

policyName

Name of the SSL policy to bind to the policy label.

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

invoke

Invoke flag.

labelType

Type of policy label invocation.

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

flowType

Flowtype of the bound SSL policy.

description

Description of the policylabel

flags**devno****count**

Example

```
i) show ssl policylabel ssl_pol_label ii) show ssl policylabel
```

ssl profile

The following operations can be performed on "ssl profile":

add | **rm** | **set** | **unset** | **show**

add ssl profile

Add a new SSL profile on the Netscaler

Synopsys

```
add ssl profile <name> [-sslProfileType ( BackEnd | FrontEnd )] [-dhCount <positive_integer>] [-dh ( ENABLED | DISABLED )] [-dhFile <string>] [-eRSA ( ENABLED | DISABLED )] [-eRSACount <positive_integer>]] [-sessReuse ( ENABLED | DISABLED )] [-sessTimeout <positive_integer>]] [-cipherRedirect ( ENABLED | DISABLED )] [-cipherURL <URL>]] [-clientAuth ( ENABLED | DISABLED )] [-clientCert ( Mandatory | Optional )]] [-sslRedirect ( ENABLED | DISABLED )] [-redirectPortRewrite ( ENABLED | DISABLED )] [-nonFipsCiphers ( ENABLED | DISABLED )] [-ssl3 ( ENABLED | DISABLED )] [-tls1 ( ENABLED | DISABLED )] [-tls11 ( ENABLED | DISABLED )] [-tls12 ( ENABLED | DISABLED )] [-SNIEnable ( ENABLED | DISABLED )] [-serverAuth ( ENABLED | DISABLED )] [-commonName <string>]] [-pushEncTrigger <pushEncTrigger>] [-sendCloseNotify ( YES | NO )] [-clearTextPort <port|*>] [-insertionEncoding ( Unicode | UTF-8 )] [-denySSLReneg <denySSLReneg>] [-quantumSize <quantumSize>] [-strictCACHecks ( YES | NO )] [-encryptTriggerPktCount <positive_integer>] [-pushFlag <positive_integer>] [-dropReqWithNoHostHeader ( YES | NO )] [-pushEncTriggerTimeout <positive_integer>] [-sslTriggerTimeout <positive_integer>] [-clientAuthUseBoundCACChain ( ENABLED | DISABLED )]
```

Arguments

name

Name of the SSL profile

sslProfileType

Type of SSL profile. FrontEnd is for front end SSL service or vserver. BackEnd is for backend SSL service.

Possible values: BackEnd, FrontEnd

Default value: FrontEnd

dhCount

Number of interactions, between the client and the NetScaler appliance, after which the DH private-public pair is regenerated. A value of zero (0) specifies infinite use (no refresh). This parameter is not applicable when configuring a backend profile.

Minimum value: 0

Maximum value: 65534

dh

State of Diffie-Hellman (DH) key exchange. This parameter is not applicable when configuring a backend profile.

Possible values: ENABLED, DISABLED

Default value: DISABLED

dhFile

Name for and, optionally, path to the PEM-format DH parameter file to be installed. /nsconfig/ssl/ is the default path. This parameter is not applicable when configuring a backend profile.

eRSA

State of Ephemeral RSA (eRSA) key exchange. Ephemeral RSA allows clients that support only export ciphers to communicate with the secure server even if the server certificate does not support export clients. The ephemeral RSA key is automatically generated when you bind an export cipher to an SSL or TCP-based

SSL virtual server or service. When you remove the export cipher, the eRSA key is not deleted. It is reused at a later date when another export cipher is bound to an SSL or TCP-based SSL virtual server or service. The eRSA key is deleted when the appliance restarts. This parameter is not applicable when configuring a backend profile.

Possible values: ENABLED, DISABLED

Default value: ENABLED

eRSACount

Refresh count for regeneration of RSA public-key and private-key pair. Zero (0) specifies infinite usage (no refresh). This parameter is not applicable when configuring a backend profile.

Minimum value: 0

Maximum value: 65534

sessReuse

State of session reuse. Establishing the initial handshake requires CPU-intensive public key encryption operations. With the ENABLED setting, session key exchange is avoided for session resumption requests received from the client.

Possible values: ENABLED, DISABLED

Default value: ENABLED

sessTimeout

Time, in seconds, for which to keep the session active. Any session resumption request received after the timeout period will require a fresh SSL handshake and establishment of a new SSL session.

Minimum value: 0

Maximum value: 4294967294

cipherRedirect

State of Cipher Redirect. If this parameter is set to ENABLED, you can configure an SSL virtual server or service to display meaningful error messages if the SSL handshake fails because of a cipher mismatch between the virtual server or service and the client. This parameter is not applicable when configuring a backend profile.

Possible values: ENABLED, DISABLED

Default value: DISABLED

cipherURL

URL of the page to which to redirect the client in case of a cipher mismatch. Typically, this page has a clear explanation of the error or an alternative location that the transaction can continue from. This parameter is not applicable when configuring a backend profile.

clientAuth

State of client authentication. In service-based SSL offload, the service terminates the SSL handshake if the SSL client does not provide a valid certificate.

This parameter is not applicable when configuring a backend profile.

Possible values: ENABLED, DISABLED

Default value: DISABLED

clientCert

Type of client authentication. If this parameter is set to MANDATORY, the appliance terminates the SSL handshake if the SSL client does not provide a valid certificate. With the OPTIONAL setting, the appliance requests a certificate from the SSL clients but proceeds with the SSL transaction even if the client presents an invalid certificate.

This parameter is not applicable when configuring a backend SSL profile.

Caution: Define proper access control policies before changing this setting to Optional.

Possible values: Mandatory, Optional

sslRedirect

State of HTTPS redirects for the SSL service.

For an SSL session, if the client browser receives a redirect message, the browser tries to connect to the new location. However, the secure SSL session breaks if the object has moved from a secure site (https://) to an unsecure site (http://). Typically, a warning message appears on the screen, prompting the user to continue or disconnect.

If SSL Redirect is ENABLED, the redirect message is automatically converted from http:// to https:// and the SSL session does not break.

This parameter is not applicable when configuring a backend profile.

Possible values: ENABLED, DISABLED

Default value: DISABLED

redirectPortRewrite

State of the port rewrite while performing HTTPS redirect. If this parameter is set to ENABLED, and the URL from the server does not contain the standard port, the port is rewritten to the standard.

Possible values: ENABLED, DISABLED

Default value: DISABLED

nonFipsCiphers

State of usage of ciphers that are not FIPS approved. Valid only for an SSL service bound with a FIPS key and certificate.

Possible values: ENABLED, DISABLED

Default value: DISABLED

ssl3

State of SSLv3 protocol support for the SSL service.

Possible values: ENABLED, DISABLED

Default value: ENABLED

tls1

State of TLSv1.0 protocol support for the SSL service.

Possible values: ENABLED, DISABLED

Default value: ENABLED

tls11

State of TLSv1.1 protocol support for the SSL service. Enabled for Front-end service on MPX-CVM platform only.

Possible values: ENABLED, DISABLED

Default value: ENABLED

tls12

State of TLSv1.2 protocol support for the SSL service. Enabled for Front-end service on MPX-CVM platform only.

Possible values: ENABLED, DISABLED

Default value: ENABLED

SNIEnable

State of the Server Name Indication (SNI) feature on the virtual server and service-based offload. SNI helps to enable SSL encryption on multiple domains on a single virtual server or service if the domains are controlled by the same organization and share the same second-level domain name. For example, *.sports.net can be used to secure domains such as login.sports.net and help.sports.net.

Possible values: ENABLED, DISABLED

Default value: DISABLED

serverAuth

State of server authentication support for the SSL Backend profile.

Possible values: ENABLED, DISABLED

Default value: DISABLED

commonName

Name to be checked against the CommonName (CN) field in the server certificate bound to the SSL server

pushEncTrigger

Trigger encryption on the basis of the PUSH flag value. Available settings function as follows:

- * ALWAYS - Any PUSH packet triggers encryption.
- * IGNORE - Ignore PUSH packet for triggering encryption.
- * MERGE - For a consecutive sequence of PUSH packets, the last PUSH packet triggers encryption.
- * TIMER - PUSH packet triggering encryption is delayed by the time defined in the set ssl parameter command or in the Change Advanced SSL Settings dialog box.

Possible values: Always, Merge, Ignore, Timer

sendCloseNotify

Enable sending SSL Close-Notify at the end of a transaction

Possible values: YES, NO

Default value: YES

clearTextPort

The clearTextPort settings.

insertionEncoding

Encoding method used to insert the subject or issuer's name in HTTP requests to servers.

Possible values: Unicode, UTF-8

Default value: Unicode

denySSLReneg

Deny renegotiation in specified circumstances. Available settings function as follows:

- * NO - Allow SSL renegotiation.
- * FRONTEND_CLIENT - Deny secure and nonsecure SSL renegotiation initiated by the client.

* FRONTEND_CLIENTSERVER - Deny secure and nonsecure SSL renegotiation initiated by the client or the NetScaler during policy-based client authentication.

* ALL - Deny all secure and nonsecure SSL renegotiation.

* NONSECURE - Deny nonsecure SSL renegotiation. Allows only clients that support RFC 5746.

Possible values: NO, FRONTEND_CLIENT, FRONTEND_CLIENTSERVER, ALL, NONSECURE

Default value: ALL

quantumSize

Amount of data to collect before the data is pushed to the crypto hardware for encryption. For large downloads, a larger quantum size better utilizes the crypto resources.

Possible values: 4096, 8192, 16384

Default value: 8192

strictCAChecks

Enable strict CA certificate checks on the appliance.

Possible values: YES, NO

Default value: NO

encryptTriggerPktCount

Maximum number of queued packets after which encryption is triggered. Use this setting for SSL transactions that send small packets from server to NetScaler.

Default value: 45

Minimum value: 10

Maximum value: 50

pushFlag

Insert PUSH flag into decrypted, encrypted, or all records. If the PUSH flag is set to a value other than 0, the buffered records are forwarded on the basis of the value of the PUSH flag. Available settings function as follows:

0 - Auto (PUSH flag is not set.)

1 - Insert PUSH flag into every decrypted record.

2 - Insert PUSH flag into every encrypted record.

3 - Insert PUSH flag into every decrypted and encrypted record.

Minimum value: 0

Maximum value: 3

dropReqWithNoHostHeader

Host header check for SNI enabled sessions. If this check is enabled and the HTTP request does not contain the host header for SNI enabled sessions, the request is dropped.

Possible values: YES, NO

Default value: NO

pushEncTriggerTimeout

PUSH encryption trigger timeout value. The timeout value is applied only if you set the Push Encryption Trigger parameter to Timer in the SSL virtual server settings.

Default value: 1

Minimum value: 1

Maximum value: 200

sslTriggerTimeout

Time, in milliseconds, after which encryption is triggered for transactions that are not tracked on the NetScaler appliance because their length is not known. There can be a delay of up to 10ms from the specified timeout value before the packet is pushed into the queue.

Default value: 100

Minimum value: 1

Maximum value: 200

clientAuthUseBoundCAChain

Certificates bound on the VIP are used for validating the client cert. Certificates came along with client cert are not used for validating the client cert

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
add sslProfile <profile name> -type front
```

rm ssl profile

Remove a SSL profile on the Netscaler

Synopsys

```
rm ssl profile <name>
```

Arguments

name

Name of the SSL profile.

Example

```
rm sslProfile <profile name>
```

set ssl profile

Set/modify SSL profile values

Synopsys

```
set ssl profile <name> [-dh ( ENABLED | DISABLED ) -dhFile <string> -dhCount <positive_integer>] [-eRSA ( ENABLED | DISABLED ) [-eRSACount <positive_integer>]] [-sessReuse ( ENABLED | DISABLED ) [-sessTimeout <positive_integer>]] [-cipherRedirect ( ENABLED | DISABLED ) [-cipherURL <URL>]] [-clientAuth ( ENABLED | DISABLED ) [-clientCert ( Mandatory | Optional )]] [-sslRedirect ( ENABLED | DISABLED )] [-redirectPortRewrite ( ENABLED | DISABLED )] [-nonFipsCiphers ( ENABLED | DISABLED )] [-ssl3 ( ENABLED | DISABLED )] [-tls1 ( ENABLED | DISABLED )] [-tls11 ( ENABLED | DISABLED )] [-tls12 ( ENABLED | DISABLED )] [-SNIEnable ( ENABLED | DISABLED )] [-serverAuth ( ENABLED | DISABLED )] [-commonName <string>]] [-pushEncTrigger <pushEncTrigger>] [-sendCloseNotify ( YES | NO )] [-clearTextPort <port*>] [-insertionEncoding ( Unicode | UTF-8 )] [-denySSLReneg <denySSLReneg>] [-quantumSize <quantumSize>] [-strictCAChecks ( YES | NO )] [-
```

encryptTriggerPktCount <positive_integer>] [-pushFlag <positive_integer>] [-dropReqWithNoHostHeader (YES | NO)] [-pushEncTriggerTimeout <positive_integer>] [-sslTriggerTimeout <positive_integer>] [-clientAuthUseBoundCAChain (ENABLED | DISABLED)]

Arguments

name

Name of the SSL profile

dh

State of Diffie-Hellman (DH) key exchange. This parameter is not applicable when configuring a backend profile.

Possible values: ENABLED, DISABLED

Default value: DISABLED

dhFile

Name for and, optionally, path to the PEM-format DH parameter file to be installed. /nsconfig/ssl/ is the default path. This parameter is not applicable when configuring a backend profile.

dhCount

Number of interactions, between the client and the NetScaler appliance, after which the DH private-public pair is regenerated. A value of zero (0) specifies infinite use (no refresh). This parameter is not applicable when configuring a backend profile.

Minimum value: 0

Maximum value: 65534

eRSA

State of Ephemeral RSA (eRSA) key exchange. Ephemeral RSA allows clients that support only export ciphers to communicate with the secure server even if the server certificate does not support export clients. The ephemeral RSA key is automatically generated when you bind an export cipher to an SSL or TCP-based SSL virtual server or service. When you remove the export cipher, the eRSA key is not deleted. It is reused at a later date when another export cipher is bound to an SSL or TCP-based SSL virtual server or service. The eRSA key is deleted when the appliance restarts. This parameter is not applicable when configuring a backend profile.

Possible values: ENABLED, DISABLED

Default value: ENABLED

eRSACount

Refresh count for regeneration of RSA public-key and private-key pair. Zero (0) specifies infinite usage (no refresh). This parameter is not applicable when configuring a backend profile.

Minimum value: 0

Maximum value: 65534

sessReuse

State of session reuse. Establishing the initial handshake requires CPU-intensive public key encryption operations. With the ENABLED setting, session key exchange is avoided for session resumption requests received from the client.

Possible values: ENABLED, DISABLED

Default value: ENABLED

sessTimeout

Time, in seconds, for which to keep the session active. Any session resumption request received after the timeout period will require a fresh SSL handshake and establishment of a new SSL session.

Minimum value: 0

Maximum value: 4294967294

cipherRedirect

State of Cipher Redirect. If this parameter is set to ENABLED, you can configure an SSL virtual server or service to display meaningful error messages if the SSL handshake fails because of a cipher mismatch between the virtual server or service and the client. This parameter is not applicable when configuring a backend profile.

Possible values: ENABLED, DISABLED

Default value: DISABLED

cipherURL

URL of the page to which to redirect the client in case of a cipher mismatch. Typically, this page has a clear explanation of the error or an alternative location that the transaction can continue from. This parameter is not applicable when configuring a backend profile.

clientAuth

State of client authentication. In service-based SSL offload, the service terminates the SSL handshake if the SSL client does not provide a valid certificate.

This parameter is not applicable when configuring a backend profile.

Possible values: ENABLED, DISABLED

Default value: DISABLED

clientCert

Type of client authentication. If this parameter is set to MANDATORY, the appliance terminates the SSL handshake if the SSL client does not provide a valid certificate. With the OPTIONAL setting, the appliance requests a certificate from the SSL clients but proceeds with the SSL transaction even if the client presents an invalid certificate.

This parameter is not applicable when configuring a backend SSL profile.

Caution: Define proper access control policies before changing this setting to Optional.

Possible values: Mandatory, Optional

sslRedirect

State of HTTPS redirects for the SSL service.

For an SSL session, if the client browser receives a redirect message, the browser tries to connect to the new location. However, the secure SSL session breaks if the object has moved from a secure site (https://) to an insecure site (http://). Typically, a warning message appears on the screen, prompting the user to continue or disconnect.

If SSL Redirect is ENABLED, the redirect message is automatically converted from http:// to https:// and the SSL session does not break.

This parameter is not applicable when configuring a backend profile.

Possible values: ENABLED, DISABLED

Default value: DISABLED

redirectPortRewrite

State of the port rewrite while performing HTTPS redirect. If this parameter is set to ENABLED, and the URL from the server does not contain the standard port, the port is rewritten to the standard.

Possible values: ENABLED, DISABLED

Default value: DISABLED

nonFipsCiphers

State of usage of ciphers that are not FIPS approved. Valid only for an SSL service bound with a FIPS key and certificate.

Possible values: ENABLED, DISABLED

Default value: DISABLED

ssl3

State of SSLv3 protocol support for the SSL service.

Possible values: ENABLED, DISABLED

Default value: ENABLED

tls1

State of TLSv1.0 protocol support for the SSL service.

Possible values: ENABLED, DISABLED

Default value: ENABLED

tls11

State of TLSv1.1 protocol support for the SSL service. Enabled for Front-end service on MPX-CVM platform only.

Possible values: ENABLED, DISABLED

Default value: ENABLED

tls12

State of TLSv1.2 protocol support for the SSL service. Enabled for Front-end service on MPX-CVM platform only.

Possible values: ENABLED, DISABLED

Default value: ENABLED

SNIEnable

State of the Server Name Indication (SNI) feature on the virtual server and service-based offload. SNI helps to enable SSL encryption on multiple domains on a single virtual server or service if the domains are controlled by the same organization and share the same second-level domain name. For example, *.sports.net can be used to secure domains such as login.sports.net and help.sports.net.

Possible values: ENABLED, DISABLED

Default value: DISABLED

serverAuth

State of server authentication support for the SSL Backend profile.

Possible values: ENABLED, DISABLED

Default value: DISABLED

commonName

Name to be checked against the CommonName (CN) field in the server certificate bound to the SSL server

pushEncTrigger

Trigger encryption on the basis of the PUSH flag value. Available settings function as follows:

- * ALWAYS - Any PUSH packet triggers encryption.
- * IGNORE - Ignore PUSH packet for triggering encryption.
- * MERGE - For a consecutive sequence of PUSH packets, the last PUSH packet triggers encryption.
- * TIMER - PUSH packet triggering encryption is delayed by the time defined in the set ssl parameter command or in the Change Advanced SSL Settings dialog box.

Possible values: Always, Merge, Ignore, Timer

sendCloseNotify

Enable sending SSL Close-Notify at the end of a transaction

Possible values: YES, NO

Default value: YES

clearTextPort

The clearTextPort settings.

insertionEncoding

Encoding method used to insert the subject or issuer's name in HTTP requests to servers.

Possible values: Unicode, UTF-8

Default value: Unicode

denySSLReneg

Deny renegotiation in specified circumstances. Available settings function as follows:

- * NO - Allow SSL renegotiation.
- * FRONTEND_CLIENT - Deny secure and nonsecure SSL renegotiation initiated by the client.
- * FRONTEND_CLIENTSERVER - Deny secure and nonsecure SSL renegotiation initiated by the client or the NetScaler during policy-based client authentication.
- * ALL - Deny all secure and nonsecure SSL renegotiation.
- * NONSECURE - Deny nonsecure SSL renegotiation. Allows only clients that support RFC 5746.

Possible values: NO, FRONTEND_CLIENT, FRONTEND_CLIENTSERVER, ALL, NONSECURE

Default value: ALL

quantumSize

Amount of data to collect before the data is pushed to the crypto hardware for encryption. For large downloads, a larger quantum size better utilizes the crypto resources.

Possible values: 4096, 8192, 16384

Default value: 8192

strictCAChecks

Enable strict CA certificate checks on the appliance.

Possible values: YES, NO

Default value: NO

encryptTriggerPktCount

Maximum number of queued packets after which encryption is triggered. Use this setting for SSL transactions that send small packets from server to NetScaler.

Default value: 45

Minimum value: 10

Maximum value: 50

pushFlag

Insert PUSH flag into decrypted, encrypted, or all records. If the PUSH flag is set to a value other than 0, the buffered records are forwarded on the basis of the value of the PUSH flag. Available settings function as follows:

0 - Auto (PUSH flag is not set.)

1 - Insert PUSH flag into every decrypted record.

2 - Insert PUSH flag into every encrypted record.

3 - Insert PUSH flag into every decrypted and encrypted record.

Minimum value: 0

Maximum value: 3

dropReqWithNoHostHeader

Host header check for SNI enabled sessions. If this check is enabled and the HTTP request does not contain the host header for SNI enabled sessions, the request is dropped.

Possible values: YES, NO

Default value: NO

pushEncTriggerTimeout

PUSH encryption trigger timeout value. The timeout value is applied only if you set the Push Encryption Trigger parameter to Timer in the SSL virtual server settings.

Default value: 1

Minimum value: 1

Maximum value: 200

sslTriggerTimeout

Time, in milliseconds, after which encryption is triggered for transactions that are not tracked on the NetScaler appliance because their length is not known. There can be a delay of up to 10ms from the specified timeout value before the packet is pushed into the queue.

Default value: 100

Minimum value: 1

Maximum value: 200

clientAuthUseBoundCAChain

Certificates bound on the VIP are used for validating the client cert. Certificates came along with client cert are not used for validating the client cert

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
set ssl profile <profile name> -tls1 ENABLED
```


unset ssl profile

Use this command to remove ssl profile settings. Refer to the set ssl profile command for meanings of the arguments.

Synopsys

```
unset ssl profile <name> [-dh] [-dhFile] [-dhCount] [-eRSA] [-eRSACount] [-sessReuse] [-sessTimeout] [-cipherRedirect] [-cipherURL] [-clientAuth] [-clientCert] [-sslRedirect] [-redirectPortRewrite] [-nonFipsCiphers] [-ssl3] [-tls1] [-tls11] [-tls12] [-SNIEnable] [-serverAuth] [-commonName] [-pushEncTrigger] [-sendCloseNotify] [-clearTextPort] [-insertionEncoding] [-denySSLReneg] [-quantumSize] [-strictCACHecks] [-encryptTriggerPktCount] [-pushFlag] [-dropReqWithNoHostHeader] [-pushEncTriggerTimeout] [-sslTriggerTimeout] [-clientAuthUseBoundCAChain]
```

show ssl profile

Display all the configured SSL profiles in the system. If a name is specified, then only that profile is shown.

Synopsys

```
show ssl profile [<name>]
```

Arguments

name

Name of the SSL profile for which to show detailed information.

Outputs

dh

State of Diffie-Hellman (DH) key exchange. This parameter is not applicable when configuring a backend profile.

dhFile

Name for and, optionally, path to the PEM-format DH parameter file to be installed. /nsconfig/ssl/ is the default path. This parameter is not applicable when configuring a backend profile.

dhCount

Number of interactions, between the client and the NetScaler appliance, after which the DH private-public pair is regenerated. A value of zero (0) specifies infinite use (no refresh). This parameter is not applicable when configuring a backend profile.

eRSA

State of Ephemeral RSA (eRSA) key exchange. Ephemeral RSA allows clients that support only export ciphers to communicate with the secure server even if the server certificate does not support export clients. The ephemeral RSA key is automatically generated when you bind an export cipher to an SSL or TCP-based SSL virtual server or service. When you remove the export cipher, the eRSA key is not deleted. It is reused at a later date when another export cipher is bound to an SSL or TCP-based SSL virtual server or service. The eRSA key is deleted when the appliance restarts. This parameter is not applicable when configuring a backend profile.

eRSACount

Refresh count for regeneration of RSA public-key and private-key pair. Zero (0) specifies infinite usage (no refresh). This parameter is not applicable when configuring a backend profile.

sessReuse

State of session reuse. Establishing the initial handshake requires CPU-intensive public key encryption operations. With the ENABLED setting, session key exchange is avoided for session resumption requests received from the client.

sessTimeout

Time, in seconds, for which to keep the session active. Any session resumption request received after the timeout period will require a fresh SSL handshake and establishment of a new SSL session.

cipherRedirect

State of Cipher Redirect. If this parameter is set to ENABLED, you can configure an SSL virtual server or service to display meaningful error messages if the SSL handshake fails because of a cipher mismatch between the virtual server or service and the client. This parameter is not applicable when configuring a backend profile.

cipherURL

URL of the page to which to redirect the client in case of a cipher mismatch. Typically, this page has a clear explanation of the error or an alternative location that the transaction can continue from. This parameter is not applicable when configuring a backend profile.

clientAuth

State of client authentication. In service-based SSL offload, the service terminates the SSL handshake if the SSL client does not provide a valid certificate.

This parameter is not applicable when configuring a backend profile.

clientCert

Type of client authentication. If this parameter is set to MANDATORY, the appliance terminates the SSL handshake if the SSL client does not provide a valid certificate. With the OPTIONAL setting, the appliance requests a certificate from the SSL clients but proceeds with the SSL transaction even if the client presents an invalid certificate.

This parameter is not applicable when configuring a backend SSL profile.

Caution: Define proper access control policies before changing this setting to Optional.

sslRedirect

State of HTTPS redirects for the SSL service.

For an SSL session, if the client browser receives a redirect message, the browser tries to connect to the new location. However, the secure SSL session breaks if the object has moved from a secure site (https://) to an unsecure site (http://). Typically, a warning message appears on the screen, prompting the user to continue or disconnect.

If SSL Redirect is ENABLED, the redirect message is automatically converted from http:// to https:// and the SSL session does not break.

This parameter is not applicable when configuring a backend profile.

redirectPortRewrite

State of the port rewrite while performing HTTPS redirect. If this parameter is set to ENABLED, and the URL from the server does not contain the standard port, the port is rewritten to the standard.

nonFipsCiphers

State of usage of ciphers that are not FIPS approved. Valid only for an SSL service bound with a FIPS key and certificate.

ssl3

State of SSLv3 protocol support for the SSL service.

tls1

State of TLSv1.0 protocol support for the SSL service.

tls11

State of TLSv1.1 protocol support for the SSL service. Enabled for Front-end service on MPX-CVM platform only.

tls12

State of TLSv1.2 protocol support for the SSL service. Enabled for Front-end service on MPX-CVM platform only.

SNIEnable

State of the Server Name Indication (SNI) feature on the virtual server and service-based offload. SNI helps to enable SSL encryption on multiple domains on a single virtual server or service if the domains are controlled by the same organization and share the same second-level domain name. For example, *.sports.net can be used to secure domains such as login.sports.net and help.sports.net.

serverAuth

State of server authentication support for the SSL Backend profile.

commonName

Name to be checked against the CommonName (CN) field in the server certificate bound to the SSL server

pushEncTrigger

Trigger encryption on the basis of the PUSH flag value. Available settings function as follows:

- * ALWAYS - Any PUSH packet triggers encryption.
- * IGNORE - Ignore PUSH packet for triggering encryption.
- * MERGE - For a consecutive sequence of PUSH packets, the last PUSH packet triggers encryption.
- * TIMER - PUSH packet triggering encryption is delayed by the time defined in the set ssl parameter command or in the Change Advanced SSL Settings dialog box.

sendCloseNotify

Enable sending SSL Close-Notify at the end of a transaction

clearTextPort

The clearTextPort settings.

insertionEncoding

Encoding method used to insert the subject or issuer's name in HTTP requests to servers.

denySSLReneg

Deny renegotiation in specified circumstances. Available settings function as follows:

- * NO - Allow SSL renegotiation.
- * FRONTEND_CLIENT - Deny secure and nonsecure SSL renegotiation initiated by the client.
- * FRONTEND_CLIENTSERVER - Deny secure and nonsecure SSL renegotiation initiated by the client or the NetScaler during policy-based client authentication.
- * ALL - Deny all secure and nonsecure SSL renegotiation.
- * NONSECURE - Deny nonsecure SSL renegotiation. Allows only clients that support RFC 5746.

quantumSize

Amount of data to collect before the data is pushed to the crypto hardware for encryption. For large downloads, a larger quantum size better utilizes the crypto resources.

strictCAChecks

Enable strict CA certificate checks on the appliance.

encryptTriggerPktCount

Maximum number of queued packets after which encryption is triggered. Use this setting for SSL transactions that send small packets from server to NetScaler.

pushFlag

Insert PUSH flag into decrypted, encrypted, or all records. If the PUSH flag is set to a value other than 0, the buffered records are forwarded on the basis of the value of the PUSH flag. Available settings function as follows:

0 - Auto (PUSH flag is not set.)

1 - Insert PUSH flag into every decrypted record.

2 - Insert PUSH flag into every encrypted record.

3 - Insert PUSH flag into every decrypted and encrypted record.

dropReqWithNoHostHeader

Host header check for SNI enabled sessions. If this check is enabled and the HTTP request does not contain the host header for SNI enabled sessions, the request is dropped.

pushEncTriggerTimeout

PUSH encryption trigger timeout value. The timeout value is applied only if you set the Push Encryption Trigger parameter to Timer in the SSL virtual server settings.

sslTriggerTimeout

Time, in milliseconds, after which encryption is triggered for transactions that are not tracked on the NetScaler appliance because their length is not known. There can be a delay of up to 10ms from the specified timeout value before the packet is pushed into the queue.

sslProfileType

Type of SSL profile. FrontEnd is for front end SSL service or vserver. BackEnd is for backend SSL service.

clientAuthUseBoundCAChain

Certificates bound on the VIP are used for validating the client cert. Certificates came along with client cert are not used for validating the client cert

devno

count

stateflag

Example

```
show ssl profile [profile name]
```

ssl rsakey

The following operations can be performed on "ssl rsakey":

create ssl rsakey

Generates an RSA key.

Synopsys

```
create ssl rsakey <keyFile> <bits> [-exponent ( 3 | F4 )] [-keyform ( DER | PEM )] [-des | -des3] {-password }
```

Arguments

keyFile

Name for and, optionally, path to the RSA key file. /nsconfig/ssl/ is the default path.

Maximum value: 63

bits

Size, in bits, of the RSA key.

Minimum value: 512

Maximum value: 4096

exponent

Public exponent for the RSA key. The exponent is part of the cipher algorithm and is required for creating the RSA key.

Possible values: 3, F4

Default value: F4

keyform

Format in which the RSA key file is stored on the appliance.

Possible values: DER, PEM

Default value: PEM

des

Encrypt the generated RSA key by using the DES algorithm. On the command line, you are prompted to enter the pass phrase (password) that is used to encrypt the key.

des3

Encrypt the generated RSA key by using the Triple-DES algorithm. On the command line, you are prompted to enter the pass phrase (password) that is used to encrypt the key.

password

Pass phrase to use for encryption if DES or DES3 option is selected.

Maximum value: 31

Example

```
create ssl rsakey /nsconfig/ssl/rsa1024.pem 1024 -exp F4
```

ssl service

The following operations can be performed on "ssl service":

set | unset | bind | unbind | show

set ssl service

Sets the advanced SSL configuration for an SSL service.

Synopsys

```
set ssl service <serviceName>@ [-dh ( ENABLED | DISABLED ) -dhFile <string>] [-dhCount <positive_integer>] [-eRSA ( ENABLED | DISABLED ) [-eRSACount <positive_integer>]] [-sessReuse ( ENABLED | DISABLED ) [-sessTimeout <positive_integer>]] [-cipherRedirect ( ENABLED | DISABLED ) [-cipherURL <URL>]] [-sslv2Redirect ( ENABLED | DISABLED ) [-sslv2URL <URL>]] [-clientAuth ( ENABLED | DISABLED ) [-clientCert ( Mandatory | Optional )]] [-sslRedirect ( ENABLED | DISABLED )] [-redirectPortRewrite ( ENABLED | DISABLED )] [-nonFipsCiphers ( ENABLED | DISABLED )] [-ssl2 ( ENABLED | DISABLED )] [-ssl3 ( ENABLED | DISABLED )] [-tls1 ( ENABLED | DISABLED )] [-tls11 ( ENABLED | DISABLED )] [-tls12 ( ENABLED | DISABLED )] [-SNIEnable ( ENABLED | DISABLED )] [-serverAuth ( ENABLED | DISABLED ) [-commonName <string>]] [-pushEncTrigger <pushEncTrigger>] [-sendCloseNotify ( YES | NO )] [-dtlsProfileName <string>] [-sslProfile <string>]
```

Arguments

serviceName

Name of the SSL service.

dh

State of Diffie-Hellman (DH) key exchange. This parameter is not applicable when configuring a backend service.

Possible values: ENABLED, DISABLED

Default value: DISABLED

dhFile

Name for and, optionally, path to the PEM-format DH parameter file to be installed. /nsconfig/ssl/ is the default path. This parameter is not applicable when configuring a backend service.

dhCount

Number of interactions, between the client and the NetScaler appliance, after which the DH private-public pair is regenerated. A value of zero (0) specifies infinite use (no refresh). This parameter is not applicable when configuring a backend service.

Minimum value: 0

Maximum value: 65534

eRSA

State of Ephemeral RSA (eRSA) key exchange. Ephemeral RSA allows clients that support only export ciphers to communicate with the secure server even if the server certificate does not support export clients. The ephemeral RSA key is automatically generated when you bind an export cipher to an SSL or TCP-based SSL virtual server or service. When you remove the export cipher, the eRSA key is not deleted. It is reused at a later date when another export cipher is bound to an SSL or TCP-based SSL virtual server or service. The eRSA key is deleted when the appliance restarts.

This parameter is not applicable when configuring a backend service.

Possible values: ENABLED, DISABLED

Default value: DISABLED

eRSACount

Refresh count for regeneration of RSA public-key and private-key pair. Zero (0) specifies infinite usage (no refresh).

This parameter is not applicable when configuring a backend service.

Minimum value: 0

Maximum value: 65534

sessReuse

State of session reuse. Establishing the initial handshake requires CPU-intensive public key encryption operations. With the ENABLED setting, session key exchange is avoided for session resumption requests received from the client.

Possible values: ENABLED, DISABLED

Default value: ENABLED

sessTimeout

Time, in seconds, for which to keep the session active. Any session resumption request received after the timeout period will require a fresh SSL handshake and establishment of a new SSL session.

Default value: 300

Minimum value: 0

Maximum value: 4294967294

cipherRedirect

State of Cipher Redirect. If this parameter is set to ENABLED, you can configure an SSL virtual server or service to display meaningful error messages if the SSL handshake fails because of a cipher mismatch between the virtual server or service and the client.

This parameter is not applicable when configuring a backend service.

Possible values: ENABLED, DISABLED

Default value: DISABLED

cipherURL

URL of the page to which to redirect the client in case of a cipher mismatch. Typically, this page has a clear explanation of the error or an alternative location that the transaction can continue from.

This parameter is not applicable when configuring a backend service.

ssl2Redirect

State of SSLv2 Redirect. If this parameter is set to ENABLED, you can configure an SSL virtual server or service to display meaningful error messages if the SSL handshake fails because of a protocol version mismatch between the virtual server or service and the client.

This parameter is not applicable when configuring a backend service.

Possible values: ENABLED, DISABLED

Default value: DISABLED

ssl2URL

URL of the page to which to redirect the client in case of a protocol version mismatch. Typically, this page has a clear explanation of the error or an alternative location that the transaction can continue from.

This parameter is not applicable when configuring a backend service.

clientAuth

State of client authentication. In service-based SSL offload, the service terminates the SSL handshake if the SSL client does not provide a valid certificate.

This parameter is not applicable when configuring a backend service.

Possible values: ENABLED, DISABLED

Default value: DISABLED

clientCert

Type of client authentication. If this parameter is set to MANDATORY, the appliance terminates the SSL handshake if the SSL client does not provide a valid certificate. With the OPTIONAL setting, the appliance requests a certificate from the SSL clients but proceeds with the SSL transaction even if the client presents an invalid certificate.

This parameter is not applicable when configuring a backend SSL service.

Caution: Define proper access control policies before changing this setting to Optional.

Possible values: Mandatory, Optional

sslRedirect

State of HTTPS redirects for the SSL service.

For an SSL session, if the client browser receives a redirect message, the browser tries to connect to the new location. However, the secure SSL session breaks if the object has moved from a secure site (https://) to an unsecure site (http://). Typically, a warning message appears on the screen, prompting the user to continue or disconnect.

If SSL Redirect is ENABLED, the redirect message is automatically converted from http:// to https:// and the SSL session does not break.

This parameter is not applicable when configuring a backend service.

Possible values: ENABLED, DISABLED

Default value: DISABLED

redirectPortRewrite

State of the port rewrite while performing HTTPS redirect. If this parameter is set to ENABLED, and the URL from the server does not contain the standard port, the port is rewritten to the standard.

Possible values: ENABLED, DISABLED

Default value: DISABLED

nonFipsCiphers

State of usage of ciphers that are not FIPS approved. Valid only for an SSL service bound with a FIPS key and certificate.

Possible values: ENABLED, DISABLED

Default value: DISABLED

ssl2

State of SSLv2 protocol support for the SSL service.

This parameter is not applicable when configuring a backend service.

Possible values: ENABLED, DISABLED

Default value: DISABLED

ssl3

State of SSLv3 protocol support for the SSL service.

Possible values: ENABLED, DISABLED

Default value: ENABLED

tls1

State of TLSv1.0 protocol support for the SSL service.

Possible values: ENABLED, DISABLED

Default value: ENABLED

tls11

State of TLSv1.1 protocol support for the SSL service. Enabled for Front-end service on MPX-CVM platform only.

Possible values: ENABLED, DISABLED

Default value: ENABLED

tls12

State of TLSv1.2 protocol support for the SSL service. Enabled for Front-end service on MPX-CVM platform only.

Possible values: ENABLED, DISABLED

Default value: ENABLED

SNIEnable

State of the Server Name Indication (SNI) feature on the virtual server and service-based offload. SNI helps to enable SSL encryption on multiple domains on a single virtual server or service if the domains are controlled by the same organization and share the same second-level domain name. For example, *.sports.net can be used to secure domains such as login.sports.net and help.sports.net.

Possible values: ENABLED, DISABLED

Default value: DISABLED

serverAuth

State of server authentication support for the SSL service.

Possible values: ENABLED, DISABLED

Default value: DISABLED

commonName

Name to be checked against the CommonName (CN) field in the server certificate bound to the SSL server

pushEncTrigger

Trigger encryption on the basis of the PUSH flag value. Available settings function as follows:

- * ALWAYS - Any PUSH packet triggers encryption.
- * IGNORE - Ignore PUSH packet for triggering encryption.
- * MERGE - For a consecutive sequence of PUSH packets, the last PUSH packet triggers encryption.
- * TIMER - PUSH packet triggering encryption is delayed by the time defined in the set ssl parameter command or in the Change Advanced SSL Settings dialog box.

Possible values: Always, Merge, Ignore, Timer

sendCloseNotify

Enable sending SSL Close-Notify at the end of a transaction

Possible values: YES, NO

Default value: YES

dtlsProfileName

Name of the DTLS profile whose settings are to be applied to the virtual server.

sslProfile

SSL profile associated to service

Example

```
1) set ssl service sslsvc -dh ENABLED -dhFile /nsconfig/ssl/dh1024.pem -dhCount 500 The al
```

unset ssl service

Use this command to remove ssl service settings. Refer to the set ssl service command for meanings of the arguments.

Synopsys

```
unset ssl service <serviceName>@ [-dh] [-dhFile] [-dhCount] [-eRSA] [-eRSACount] [-sessReuse] [-sessTimeout] [-cipherRedirect] [-cipherURL] [-sslv2Redirect] [-sslv2URL] [-clientAuth] [-clientCert] [-sslRedirect] [-redirectPortRewrite] [-nonFipsCiphers] [-ssl2] [-ssl3] [-tls1] [-tls11] [-tls12] [-SNIEnable] [-serverAuth] [-commonName] [-sendCloseNotify] [-dtlsProfileName] [-sslProfile]
```

bind ssl service

Binds an SSL certificate-key pair or an SSL policy to a transparent SSL service.

Synopsys

```
bind ssl service <serviceName>@ ((-policyName <string> [-priority <positive_integer>] [-gotoPriorityExpression <expression>] [-invoke (<labelType> <labelName>)] ) | ((-certkeyName <string> [(-CA [-crlCheck ( Mandatory | Optional )] -ocspCheck ( Mandatory | Optional ))] [-skipCAName]) | -SNICert] ) | -cipherName <string> | -eccCurveName <eccCurveName>))
```

Arguments

serviceName

Name of the SSL service for which to set advanced configuration.

policyName

Name of the SSL policy to bind to the service.

priority

Priority.

Minimum value: 0

Maximum value: 64000

gotoPriorityExpression

Expression or other value specifying the next policy to be evaluated if the current policy evaluates to TRUE. Specify one of the following values:

* NEXT - Evaluate the policy with the next higher priority number.

* END - End policy evaluation.

* **USE_INVOCATION_RESULT** - Applicable if this policy invokes another policy label. If the final goto in the invoked policy label has a value of **END**, the evaluation stops. If the final goto is anything other than **END**, the current policy label performs a **NEXT**.

* A default syntax or classic expression that evaluates to a number.

If you specify an expression, the number to which it evaluates determines the next policy to evaluate, as follows:

* If the expression evaluates to a higher numbered priority, the policy with that priority is evaluated next.

* If the expression evaluates to the priority of the current policy, the policy with the next higher numbered priority is evaluated next.

* If the expression evaluates to a number that is larger than the largest numbered priority, policy evaluation ends.

An **UNDEF** event is triggered if:

* The expression is invalid.

* The expression evaluates to a priority number that is numerically lower than the current policy's priority.

* The expression evaluates to a priority number that is between the current policy's priority number (say, 30) and the highest priority number (say, 100), but does not match any configured priority number (for example, the expression evaluates to the number 85). This example assumes that the priority number increments by 10 for every successive policy, and therefore a priority number of 85 does not exist in the policy label.

Default value: "END"

invoke

Invoke policies bound to a virtual server, service, or policy label. After the invoked policies are evaluated, the flow returns to the policy with the next-larger priority number.

labelType

Type of policy label invocation.

Possible values: vserver, service, policylabel

labelName

Name of the policy label, virtual server, or service to invoke if the current policy rule evaluates to **TRUE**.

certkeyName

Name of the certificate-key pair.

CA

Name of the CA certificate that issues and signs the intermediate-CA certificate or the end-user client or server certificate.

crlCheck

Rule to use for the CRL corresponding to the CA certificate during client authentication. Available settings function as follows:

* **MANDATORY** - Deny SSL clients if the CRL is missing or expired, or the Next Update date is in the past, or the CRL is incomplete.

* **OPTIONAL** - Allow SSL clients if the CRL is missing or expired, or the Next Update date is in the past, or the CRL is incomplete, but deny if the client certificate is revoked in the CRL.

Possible values: Mandatory, Optional

skipCAName

The flag is used to indicate whether this particular CA certificate's **CA_Name** needs to be sent to the SSL client while requesting for client certificate in a SSL handshake

SNICert

Name of the certificate-key pair to bind for use in SNI processing.

ocspCheck

Rule to use for the OCSP responder associated with the CA certificate during client authentication. If MANDATORY is specified, deny all SSL clients if the OCSP check fails because of connectivity issues with the remote OCSP server, or any other reason that prevents the OCSP check. With the OPTIONAL setting, allow SSL clients even if the OCSP check fails except when the client certificate is revoked.

Possible values: Mandatory, Optional

cipherName

Name of the individual cipher, user-defined cipher group, or predefined (built-in) cipher alias.

eccCurveName

Named ECC curve bound to service/vserver.

Possible values: ALL, P_224, P_256, P_384, P_521

Example

```
bind ssl service ssl_svc -policyName certInsert_pol -priority 10
```

unbind ssl service

Unbinds an SSL policy, cipher, and certificate-key pair from an SSL service.

Synopsys

```
unbind ssl service <serviceName>@ ((-policyName <string> [-priority <positive_integer>]) | ((-certkeyName <string>  
[(-CA [-crlCheck ( Mandatory | Optional )]) | -SNICert ] ) | -cipherName <string> | -eccCurveName  
<eccCurveName>))
```

Arguments

serviceName

Name of the SSL service.

policyName

Name of the SSL policy to unbind from the SSL service.

priority

Priority of the NOPOLICY (built-in policy) to be unbound. Not required if you are unbinding a user-defined policy.

Minimum value: 1

Maximum value: 2147483647

certkeyName

The certificate key pair binding.

CA

CA certificate.

crlCheck

Rule to use for the CRL corresponding to the CA certificate during client authentication. Available settings function as follows:

* MANDATORY - Deny SSL clients if the CRL is missing or expired, or the Next Update date is in the past, or the CRL is incomplete.

* OPTIONAL - Allow SSL clients if the CRL is missing or expired, or the Next Update date is in the past, or the CRL is incomplete, but deny if the client certificate is revoked in the CRL.

Possible values: Mandatory, Optional

Default value: CRLCHECK_OPTIONAL

SNICert

Name of the certificate-key pair to bind for use in SNI processing.

cipherName

Name of the individual cipher, user-defined cipher group, or predefined (built-in) cipher alias.

eccCurveName

Named ECC curve bound to service/vserver.

Possible values: ALL, P_224, P_256, P_384, P_521

Example

```
unbind ssl service ssl_svc -policyName certInsert_pol
```

show ssl service

Displays information about SSL-specific configuration information for all SSL services, or displays detailed information about the specified SSL service.

Synopsis

```
show ssl service [<serviceName>] [-cipherDetails]
```

Arguments

serviceName

Name of the SSL service for which to show detailed information.

cipherDetails

Display details of the individual ciphers bound to the SSL service.

Outputs

crlCheck

The state of the CRL check parameter. (Mandatory/Optional)

dh

The state of Diffie-Hellman (DH) key exchange support.

dhFile

The file name and path for the DH parameter.

dhCount

The refresh count for regeneration of DH public-key and private-key from the DH parameter.

eRSA

The state of Ephemeral RSA key exchange support. Ephemeral RSA is used for export ciphers

eRSACount

The refresh count for re-generation of RSA public-key and pri-vate-key pair.

sessReuse

The state of session reuse support.

sessTimeout

The session timeout value in seconds.

cipherRedirect

The state of Cipher Redirect feature. Cipher Redirect feature can be used to provide more readable information to SSL clients about mismatch in ciphers between the client and the SSL vserver.

cipherURL

The redirect URL to be used with the Cipher Redirect feature.

ssl2Redirect

The state of SSLv2 Redirect feature. SSLv2 Redirect feature can be used to provide more readable information to SSL client about non-support of SSLv2 protocol on the SSL vserver.

ssl2URL

The redirect URL to be used with the SSLv2 Redirect feature.

clientAuth

The state of Client-Authentication support.

clientCert

The rule for client certificate requirement in client authentication.

sslRedirect

The state of HTTPS redirect feature.

redirectPortRewrite

The state of port rewrite feature.

nonFipsCiphers

The state of usage of non FIPS approved ciphers.

ssl2

The state of SSLv2 protocol support.

ssl3

The state of SSLv3 protocol support.

tls1

The state of TLSv1.0 protocol support.

tls11

The state of TLSv1.1 protocol support.

tls12

The state of TLSv1.2 protocol support.

SNIEnable

The state of SNI extension. Server Name Indication (SNI) helps to enable SSL encryption on multiple subdomains if the domains are controlled by the same organization and share the same second-level domain name.

serverAuth

The state of Server-Authentication support.

commonName

Name to be checked against the CommonName (CN) field in the server certificate bound to the SSL server

cipherAliasName/cipherName/cipherGroupName

The cipher group/alias/individual cipher configuration.

cipherName

The cipher group/alias/individual cipher configuration

description

The cipher suite description.

certkeyName

The certificate key pair binding.

policyName

The SSL policy binding.

invoke

Invoke flag. This attribute is relevant only for ADVANCED policies

labelType

Type of policy label invocation.

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

clearTextPort

The clearTextPort settings.

service**priority**

The priority of the policies bound to this SSL service

polinherit

Whether the bound policy is a inherited policy or not

ocspCheck

Rule to use for the OCSP responder associated with the CA certificate during client authentication. If MANDATORY is specified, deny all SSL clients if the OCSP check fails because of connectivity issues with the remote OCSP server, or any other reason that prevents the OCSP check. With the OPTIONAL setting, allow SSL clients even if the OCSP check fails except when the client certificate is revoked.

pushEncTrigger

PUSH packet triggering encryption: Always, Ignore, Merge

CA

CA certificate.

SNICert

The name of the CertKey. Use this option to bind Certkey(s) which will be used in SNI processing.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

stateflag

skipCAName

The flag is used to indicate whether this particular CA certificate's CA_Name needs to be sent to the SSL client while requesting for client certificate in a SSL handshake

sendCloseNotify

Enable sending SSL Close-Notify at the end of a transaction

dtlsProfileName

Name of the DTLS profile whose settings are to be applied to the virtual server.

dtlsFlag

The flag is used to indicate whether DTLS is set or not

eccCurveName

Named ECC curve bound to service/vserver.

sslProfile

SSL profile associated to service

devno

count

Example

An example of output of show ssl service command is as shown below show ssl service svc1

ssl serviceGroup

The following operations can be performed on "ssl serviceGroup":

set | **unset** | **bind** | **unbind** | **show**

set ssl serviceGroup

Sets the advanced SSL configuration for an SSL service group.

Synopsys

```
set ssl serviceGroup <serviceName>@ [-sslProfile <string>] [-sessReuse ( ENABLED | DISABLED ) [-  
sessTimeout <positive_integer>]] [-nonFipsCiphers ( ENABLED | DISABLED )] [-ssl3 ( ENABLED | DISABLED )] [-  
tls1 ( ENABLED | DISABLED )] [-tls11 ( ENABLED | DISABLED )] [-tls12 ( ENABLED | DISABLED )] [-serverAuth (   
ENABLED | DISABLED ) [-commonName <string>]] [-sendCloseNotify ( YES | NO )]
```

Arguments

serviceName

Name of the SSL service group for which to set advanced configuration.

sslProfile

SSL Profile associated to serviceGroup

sessReuse

State of session reuse. Establishing the initial handshake requires CPU-intensive public key encryption operations. With the ENABLED setting, session key exchange is avoided for session resumption requests received from the client.

Possible values: ENABLED, DISABLED

Default value: ENABLED

sessTimeout

Time, in seconds, for which to keep the session active. Any session resumption request received after the timeout period will require a fresh SSL handshake and establishment of a new SSL session.

Default value: 300

Minimum value: 0

Maximum value: 4294967294

nonFipsCiphers

State of usage of ciphers that are not FIPS approved. Valid only for an SSL service bound with a FIPS key and certificate.

Possible values: ENABLED, DISABLED

Default value: DISABLED

ssl3

State of SSLv3 protocol support for the SSL service group.

Possible values: ENABLED, DISABLED

Default value: ENABLED

tls1

State of TLSv1.0 protocol support for the SSL service group.

Possible values: ENABLED, DISABLED

Default value: ENABLED

tls11

State of TLSv1.1 protocol support for the SSL service group.

Possible values: ENABLED, DISABLED

Default value: ENABLED

tls12

State of TLSv1.2 protocol support for the SSL service group.

Possible values: ENABLED, DISABLED

Default value: ENABLED

serverAuth

State of server authentication support for the SSL service group.

Possible values: ENABLED, DISABLED

Default value: DISABLED

commonName

Name to be checked against the CommonName (CN) field in the server certificate bound to the SSL server

sendCloseNotify

Enable sending SSL Close-Notify at the end of a transaction

Possible values: YES, NO

Default value: YES

Example

```
1) set ssl servicegroup svcg1 -sessReuse DISABLED The above example disables session reuse
```

unset ssl serviceGroup

Use this command to remove ssl serviceGroup settings. Refer to the set ssl serviceGroup command for meanings of the arguments.

Synopsis

```
unset ssl serviceGroup <serviceName>@ [-sslProfile] [-sessReuse] [-sessTimeout] [-nonFipsCiphers] [-ssl3] [-tls1] [-tls11] [-tls12] [-serverAuth] [-commonName] [-sendCloseNotify]
```

bind ssl serviceGroup

Bind a SSL certkey or a SSL policy to a SSL service.

Synopsis

```
bind ssl serviceGroup <serviceName>@ ((-certkeyName <string> [(-CA [-crlCheck ( Mandatory | Optional ) | -ocspCheck ( Mandatory | Optional )]) | -SNICert] ) | -cipherName <string>)
```

Arguments

serviceName

The name of the SSL service to which the SSL policy needs to be bound.

certkeyName

The name of the CertKey

CA

CA certificate.

crlCheck

The rule for use of CRL corresponding to this CA certificate during client authentication. If crlCheck is set to Mandatory, the system will deny all SSL clients if the CRL is missing, expired - NextUpdate date is in the past, or is incomplete with remote CRL refresh enabled. If crlCheck is set to optional, the system will allow SSL clients in the above error cases. However, in any case if the client certificate is revoked in the CRL, the SSL client will be denied access.

Possible values: Mandatory, Optional

SNICert

The name of the CertKey. Use this option to bind Certkey(s) which will be used in SNI processing.

ocspCheck

The state of the OCSP check parameter. (Mandatory/Optional)

Possible values: Mandatory, Optional

cipherName

A cipher-suite can consist of an individual cipher name, the system predefined cipher-alias name, or user defined cipher-group name.

Example

```
bind ssl service ssl_svc -policyName certInsert_pol -priority 10
```

unbind ssl serviceGroup

Unbind a SSL policy from a SSL service.

Synopsys

```
unbind ssl serviceGroup <serviceGroupName>@ ((-certkeyName <string> [(-CA [-crlCheck ( Mandatory | Optional )]) | -SNICert] ) | -cipherName <string>)
```

Arguments

serviceGroupName

The name of the SSL service from which the SSL policy needs to be unbound.

certkeyName

The name of the certificate bound to the SSL service group.

CA

CA certificate.

crlCheck

The rule for use of CRL corresponding to this CA certificate during client authentication. If crlCheck is set to Mandatory, the system will deny all SSL clients if the CRL is missing, expired - NextUpdate date is in the

past, or is incomplete with remote CRL refresh enabled. If `crlCheck` is set to optional, the system will allow SSL clients in the above error cases. However, in any case if the client certificate is revoked in the CRL, the SSL client will be denied access.

Possible values: Mandatory, Optional

SNICert

The name of the CertKey. Use this option to bind Certkey(s) which will be used in SNI processing.

cipherName

A cipher-suite can consist of an individual cipher name, the system predefined cipher-alias name, or user defined cipher-group name.

Example

```
unbind ssl service ssl_svc -policyName certInsert_pol
```

show ssl serviceGroup

Displays information about SSL-specific configuration for all SSL service groups, or displays detailed information about the specified SSL service group.

Synopsys

```
show ssl serviceGroup [<serviceName>] [-cipherDetails]
```

Arguments

serviceName

Name of the SSL service group for which to show detailed information.

cipherDetails

Display details of the individual ciphers bound to the SSL service group.

Outputs

dh

The state of DH key exchange support for the SSL service group.

dhFile

The file name and path for the DH parameter.

dhCount

The refresh count for the re-generation of DH public-key and private-key from the DH parameter.

eRSA

The state of Ephemeral RSA key exchange support for the SSL service group. Ephemeral RSA is used for export ciphers.

eRSACount

The refresh count for the re-generation of RSA public-key and private-key pair.

sessReuse

The state of session re-use support for the SSL service group.

sessTimeout

The Session timeout value in seconds.

cipherRedirect

The state of Cipher Redirect feature. Cipher Redirect feature can be used to provide more readable information to SSL clients about mismatch in ciphers between the client and the SSL vserver.

cipherURL

The redirect URL to be used with the Cipher Redirect feature.

ssl2Redirect

The state of SSLv2 Redirect feature. SSLv2 Redirect feature can be used to provide more readable information to SSL client about non-support of SSLv2 protocol on the SSL vserver.

ssl2URL

The redirect URL to be used with SSLv2 Redirect feature.

clientAuth

The state of Client-Authentication support for the SSL service group.

clientCert

The rule for client certificate requirement in client authentication.

sslRedirect

The state of HTTPS redirects for the SSL service group.

This is required for the proper functioning of the redirect messages from the server. The redirect message from the server provides the new location for the moved object. This is contained in the HTTP header field: Location, e.g. Location: <http://www.moved.org/here.html>

For the SSL session, if the client browser receives this message, the browser will try to connect to the new location. This will break the secure SSL session, as the object has moved from a secure site (<https://>) to an un-secure one (<http://>). Generally browsers flash a warning message on the screen and prompt the user, either to continue or disconnect.

The above feature, when enabled will automatically convert all such <http://> redirect message to <https://>. This will not break the client SSL session.

Note: The set ssl service command can be used for configuring a front-end SSL service for service based SSL Off-Loading, or a backend SSL service for backend-encryption setup.

redirectPortRewrite

The state of port-rewrite feature.

nonFipsCiphers

The state of usage of non FIPS approved ciphers.

ssl2

The state of SSLv2 protocol support for the SSL service group.

ssl3

State of SSLv3 protocol support for the SSL service group.

tls1

State of TLSv1.0 protocol support for the SSL service group.

tls11

State of TLSv1.1 protocol support for the SSL service group.

tls12

State of TLSv1.2 protocol support for the SSL service group.

serverAuth

The state of the server authentication configuration for the SSL service group. For SSL deployments where data is encrypted end-to-end using SSL, you can authenticate the server.

commonName

Name to be checked against the CommonName (CN) field in the server certificate bound to the SSL server

cipherAliasName/cipherName/cipherGroupName

The name of the cipher group/alias/name configured for the SSL service group.

cipherName

The name of the cipher group/alias/name configured for the SSL service group.

ocspCheck

The state of the OCSP check parameter. (Mandatory/Optional)

crlCheck

The state of the CRL check parameter. (Mandatory/Optional)

description

The description of the cipher.

certkeyName

The name of the certificate bound to the SSL service group.

clearTextPort

The port on the back-end web-servers where the clear-text data is sent by system. Use this setting for the wildcard IP based SSL Acceleration configuration (*:443).

serviceName

The service name.

CA

CA certificate.

SNICert

The name of the CertKey. Use this option to bind Certkey(s) which will be used in SNI processing.

stateflag

sendCloseNotify

Enable sending SSL Close-Notify at the end of a transaction

sslProfile

SSL Profile associated to serviceGroup

devno

count

Example

An example of output of show ssl servicegroup command is as shown below show ssl serviceg:

ssl stats

The following operations can be performed on "ssl stats":

show ssl stats

show ssl stats is an alias for stat ssl

Synopsys

show ssl stats - alias for 'stat ssl'

ssl vserver

The following operations can be performed on "ssl vserver":

[set](#) | [unset](#) | [bind](#) | [unbind](#) | [show](#)

set ssl vserver

Sets advanced SSL configuration for an SSL virtual server.

Synopsys

```
set ssl vserver <vServerName>@ [-clearTextPort <port>] [-dh ( ENABLED | DISABLED ) -dhFile <string>] [-dhCount <positive_integer>] [-eRSA ( ENABLED | DISABLED ) [-eRSACount <positive_integer>]] [-sessReuse ( ENABLED | DISABLED ) [-sessTimeout <positive_integer>]] [-cipherRedirect ( ENABLED | DISABLED ) [-cipherURL <URL>]] [-sslv2Redirect ( ENABLED | DISABLED ) [-sslv2URL <URL>]] [-clientAuth ( ENABLED | DISABLED ) [-clientCert ( Mandatory | Optional )]] [-sslRedirect ( ENABLED | DISABLED )] [-redirectPortRewrite ( ENABLED | DISABLED )] [-nonFipsCiphers ( ENABLED | DISABLED )] [-ssl2 ( ENABLED | DISABLED )] [-ssl3 ( ENABLED | DISABLED )] [-tls1 ( ENABLED | DISABLED )] [-tls11 ( ENABLED | DISABLED )] [-tls12 ( ENABLED | DISABLED )] [-SNIEnable ( ENABLED | DISABLED )] [-pushEncTrigger <pushEncTrigger>] [-sendCloseNotify ( YES | NO )] [-dtlsProfileName <string>] [-sslProfile <string>]
```

Arguments

vServerName

Name of the SSL virtual server for which to set advanced configuration.

clearTextPort

Port on which clear-text data is sent by the appliance to the server. Do not specify this parameter for SSL offloading with end-to-end encryption.

Default value: 0

dh

State of Diffie-Hellman (DH) key exchange.

Possible values: ENABLED, DISABLED

Default value: DISABLED

dhFile

Name of and, optionally, path to the DH parameter file, in PEM format, to be installed. /nsconfig/ssl/ is the default path.

dhCount

Number of interactions, between the client and the NetScaler appliance, after which the DH private-public pair is regenerated. A value of zero (0) specifies infinite use (no refresh).

Minimum value: 0

Maximum value: 65534

eRSA

State of Ephemeral RSA (eRSA) key exchange. Ephemeral RSA allows clients that support only export ciphers to communicate with the secure server even if the server certificate does not support export clients. The ephemeral RSA key is automatically generated when you bind an export cipher to an SSL or TCP-based SSL virtual server or service. When you remove the export cipher, the eRSA key is not deleted. It is reused at a later date when another export cipher is bound to an SSL or TCP-based SSL virtual server or service. The eRSA key is deleted when the appliance restarts.

Possible values: ENABLED, DISABLED

Default value: ENABLED

eRSACount

Refresh count for regeneration of the RSA public-key and private-key pair. Zero (0) specifies infinite usage (no refresh).

Minimum value: 0

Maximum value: 65534

sessReuse

State of session reuse. Establishing the initial handshake requires CPU-intensive public key encryption operations. With the ENABLED setting, session key exchange is avoided for session resumption requests received from the client.

Possible values: ENABLED, DISABLED

Default value: ENABLED

sessTimeout

Time, in seconds, for which to keep the session active. Any session resumption request received after the timeout period will require a fresh SSL handshake and establishment of a new SSL session.

Default value: 120

Minimum value: 0

Maximum value: 4294967294

cipherRedirect

State of Cipher Redirect. If cipher redirect is enabled, you can configure an SSL virtual server or service to display meaningful error messages if the SSL handshake fails because of a cipher mismatch between the virtual server or service and the client.

Possible values: ENABLED, DISABLED

Default value: DISABLED

cipherURL

The redirect URL to be used with the Cipher Redirect feature.

ssl2Redirect

State of SSLv2 Redirect. If SSLv2 redirect is enabled, you can configure an SSL virtual server or service to display meaningful error messages if the SSL handshake fails because of a protocol version mismatch between the virtual server or service and the client.

Possible values: ENABLED, DISABLED

Default value: DISABLED

ssl2URL

URL of the page to which to redirect the client in case of a protocol version mismatch. Typically, this page has a clear explanation of the error or an alternative location that the transaction can continue from.

clientAuth

State of client authentication. If client authentication is enabled, the virtual server terminates the SSL handshake if the SSL client does not provide a valid certificate.

Possible values: ENABLED, DISABLED

Default value: DISABLED

clientCert

Type of client authentication. If this parameter is set to MANDATORY, the appliance terminates the SSL handshake if the SSL client does not provide a valid certificate. With the OPTIONAL setting, the appliance requests a certificate from the SSL clients but proceeds with the SSL transaction even if the client presents an invalid certificate.

Caution: Define proper access control policies before changing this setting to Optional.

Possible values: Mandatory, Optional

sslRedirect

State of HTTPS redirects for the SSL virtual server.

For an SSL session, if the client browser receives a redirect message, the browser tries to connect to the new location. However, the secure SSL session breaks if the object has moved from a secure site (https://) to an unsecure site (http://). Typically, a warning message appears on the screen, prompting the user to continue or disconnect.

If SSL Redirect is ENABLED, the redirect message is automatically converted from http:// to https:// and the SSL session does not break.

Possible values: ENABLED, DISABLED

Default value: DISABLED

redirectPortRewrite

State of the port rewrite while performing HTTPS redirect. If this parameter is ENABLED and the URL from the server does not contain the standard port, the port is rewritten to the standard.

Possible values: ENABLED, DISABLED

Default value: DISABLED

nonFipsCiphers

State of usage of non-FIPS approved ciphers. Valid only for an SSL service bound with a FIPS key and certificate.

Possible values: ENABLED, DISABLED

Default value: DISABLED

ssl2

State of SSLv2 protocol support for the SSL Virtual Server.

Possible values: ENABLED, DISABLED

Default value: DISABLED

ssl3

State of SSLv3 protocol support for the SSL Virtual Server.

Possible values: ENABLED, DISABLED

Default value: ENABLED

tls1

State of TLSv1.0 protocol support for the SSL Virtual Server.

Possible values: ENABLED, DISABLED

Default value: ENABLED

tls11

State of TLSv1.1 protocol support for the SSL Virtual Server. TLSv1.1 protocol is supported only on the MPX appliance. Support is not available on a FIPS appliance or on a NetScaler VPX virtual appliance. On an SDX appliance, TLSv1.1 protocol is supported only if an SSL chip is assigned to the instance.

Possible values: ENABLED, DISABLED

Default value: ENABLED

tls12

State of TLSv1.2 protocol support for the SSL Virtual Server. TLSv1.2 protocol is supported only on the MPX appliance. Support is not available on a FIPS appliance or on a NetScaler VPX virtual appliance. On an SDX appliance, TLSv1.2 protocol is supported only if an SSL chip is assigned to the instance.

Possible values: ENABLED, DISABLED

Default value: ENABLED

SNIEnable

State of the Server Name Indication (SNI) feature on the virtual server and service-based offload. SNI helps to enable SSL encryption on multiple domains on a single virtual server or service if the domains are controlled by the same organization and share the same second-level domain name. For example, *.sports.net can be used to secure domains such as login.sports.net and help.sports.net.

Possible values: ENABLED, DISABLED

Default value: DISABLED

pushEncTrigger

Trigger encryption on the basis of the PUSH flag value. Available settings function as follows:

- * ALWAYS - Any PUSH packet triggers encryption.
- * IGNORE - Ignore PUSH packet for triggering encryption.
- * MERGE - For a consecutive sequence of PUSH packets, the last PUSH packet triggers encryption.
- * TIMER - PUSH packet triggering encryption is delayed by the time defined in the set ssl parameter command or in the Change Advanced SSL Settings dialog box.

Possible values: Always, Merge, Ignore, Timer

sendCloseNotify

Enable sending SSL Close-Notify at the end of a transaction

Possible values: YES, NO

Default value: YES

dtlsProfileName

Name of the DTLS profile whose settings are to be applied to the virtual server.

sslProfile

SSL profile associated to vserver

Example

```
1) set ssl vserver sslvip -dh ENABLED -dhFile /siteA/dh1024.pem -dhCount 500
```

 The above ex:

unset ssl vserver

Use this command to remove ssl vserver settings. Refer to the set ssl vserver command for meanings of the arguments.

Synopsys

```
unset ssl vserver <vServerName>@ [-clearTextPort] [-dh] [-dhFile] [-dhCount] [-eRSA] [-eRSACount] [-sessReuse] [-sessTimeout] [-cipherRedirect] [-cipherURL] [-sslv2Redirect] [-sslv2URL] [-clientAuth] [-clientCert] [-sslRedirect] [-redirectPortRewrite] [-nonFipsCiphers] [-ssl2] [-ssl3] [-tls1] [-tls11] [-tls12] [-SNIEnable] [-sendCloseNotify] [-dtlsProfileName] [-sslProfile]
```

bind ssl vserver

Binds an SSL certificate-key pair or an SSL policy to an SSL virtual server.

Synopsys

```
bind ssl vserver <vServerName>@ ((-policyName <string> [-priority <positive_integer>] [-gotoPriorityExpression <expression>] [-invoke (<labelType> <labelName>) ] ) | ((-certkeyName <string> [(-CA [-crlCheck ( Mandatory | Optional ) ] -ocspCheck ( Mandatory | Optional )] [-skipCAName]) | -SNICert] ) | -cipherName <string> | -eccCurveName <eccCurveName>))
```

Arguments

vServerName

Name of the SSL virtual server.

policyName

Name of the SSL policy to bind to the SSL virtual server.

priority

Integer specifying the policy's priority. The lower the number, the higher the priority.

Minimum value: 0

Maximum value: 64000

gotoPriorityExpression

Expression or other value specifying the next policy to be evaluated if the current policy evaluates to TRUE. Specify one of the following values:

- * NEXT - Evaluate the policy with the next higher priority number.
- * END - End policy evaluation.
- * USE_INVOCATION_RESULT - Applicable if this policy invokes another policy label. If the final goto in the invoked policy label has a value of END, the evaluation stops. If the final goto is anything other than END, the current policy label performs a NEXT.
- * A default syntax or classic expression that evaluates to a number.

If you specify an expression, the number to which it evaluates determines the next policy to evaluate, as follows:

- * If the expression evaluates to a higher numbered priority, the policy with that priority is evaluated next.
- * If the expression evaluates to the priority of the current policy, the policy with the next higher numbered priority is evaluated next.
- * If the expression evaluates to a number that is larger than the largest numbered priority, policy evaluation ends.

An UNDEF event is triggered if:

- * The expression is invalid.
- * The expression evaluates to a priority number that is numerically lower than the current policy's priority.

* The expression evaluates to a priority number that is between the current policy's priority number (say, 30) and the highest priority number (say, 100), but does not match any configured priority number (for example, the expression evaluates to the number 85). This example assumes that the priority number increments by 10 for every successive policy, and therefore a priority number of 85 does not exist in the policy label.

Default value: "END"

invoke

Invoke policies bound to a virtual server, service, or user-defined policy label. After the invoked policies are evaluated, the flow returns to the policy with the next priority.

labelType

Type of policy label to invoke.

Possible values: vserver, service, policylabel

labelName

Name of the policy label, virtual server, or service to invoke if the current policy rule evaluates to TRUE.

certkeyName

Name of the certificate-key pair.

CA

Name of the CA certificate that issues and signs the intermediate-CA certificate or the end-user client or server certificate.

crlCheck

Rule to use for the CRL corresponding to the CA certificate during client authentication. Available settings function as follows:

* MANDATORY - Deny SSL clients if the CRL is missing or expired, or the Next Update date is in the past, or the CRL is incomplete.

* OPTIONAL - Allow SSL clients if the CRL is missing or expired, or the Next Update date is in the past, or the CRL is incomplete, but deny if the client certificate is revoked in the CRL.

Possible values: Mandatory, Optional

Default value: CRLCHECK_OPTIONAL

skipCAName

The flag is used to indicate whether this particular CA certificates CA Name needs to be sent to the SSL client while requesting for client certificate in a SSL handshake

SNICert

Name of the certificate-key pair to bind for use in SNI processing.

ocspCheck

Rule to use for the OCSP responder associated with the CA certificate during client authentication. If MANDATORY is specified, deny all SSL clients if the OCSP check fails because of connectivity issues with the remote OCSP server, or any other reason that prevents the OCSP check. With the OPTIONAL setting, allow SSL clients even if the OCSP check fails except when the client certificate is revoked.

Possible values: Mandatory, Optional

cipherName

Name of the individual cipher, user-defined cipher group, or predefined (built-in) cipher alias.

eccCurveName

Named ECC curve bound to service/vserver.

Possible values: ALL, P_224, P_256, P_384, P_521

Example

```
1. bind ssl vserver ssl_vip -certkeyName cert1 In the above example the certificate cert1
```

unbind ssl vserver

Unbinds an SSL policy, cipher, and certificate-key pair from an SSL virtual server.

Synopsys

```
unbind ssl vserver <vServerName>@ ((-policyName <string> [-priority <positive_integer>]) | ((-certkeyName  
<string> [-CA | -SNICert] ) | -cipherName <string> | -eccCurveName <eccCurveName>))
```

Arguments

vServerName

Name of the SSL virtual server.

policyName

Name of the SSL policy to unbind from the SSL virtual server.

priority

Priority of the NOPOLICY (built-in policy) to be unbound. Not required if you are unbinding a user-defined policy.

Minimum value: 1

Maximum value: 2147483647

certkeyName

The name of the certificate key pair binding.

CA

CA certificate.

SNICert

Name of the SNI certificate-key pair.

cipherName

Name of the cipher.

eccCurveName

Named ECC curve bound to service/vserver.

Possible values: ALL, P_224, P_256, P_384, P_521

Example

```
unbind ssl vserver ssl_vip -policyName certInsert_pol
```

show ssl vserver

Displays SSL specific configuration information for all SSL virtual servers, or displays detailed information for the specified SSL virtual server.

Synopsys

show ssl vserver [<vServerName>] [-cipherDetails]

Arguments

vServerName

Name of the SSL virtual server for which to show detailed information.

cipherDetails

Display details of the individual ciphers bound to the SSL virtual server.

Outputs

clearTextPort

The clearTextPort settings.

dh

The state of Diffie-Hellman (DH) key exchange support.

dhFile

The file name and path for the DH parameter.

dhCount

The refresh count for the re-generation of DH public-key and private-key from the DH parameter.

eRSA

The state of Ephemeral RSA key exchange support. Ephemeral RSA is used for export ciphers

eRSACount

The refresh count for the re-generation of RSA public-key and private-key pair.

sessReuse

The state of session re-use support.

sessTimeout

The Session timeout value in seconds.

cipherRedirect

The state of Cipher Redirect feature. Cipher Redirect feature can be used to provide more readable information to SSL clients about mismatch in ciphers between the client and the SSL vserver.

crlCheck

The state of the CRL check parameter. (Mandatory/Optional)

cipherURL

The redirect URL to be used with the Cipher Redirect feature.

ssl2Redirect

The state of SSLv2 Redirect feature. SSLv2 Redirect feature can be used to provide more readable information to SSL client about non-support of SSLv2 protocol on the SSL vserver.

ssl2URL

The redirect URL to be used with SSLv2 Redirect feature.

clientAuth

The state of Client-Authentication support.

clientCert

The rule for client certificate requirement in client authentication.

sslRedirect

The state of HTTPS redirect feature support.

priority

The priority of the policies bound to this SSL service

polinherit

Whether the bound policy is a inherited policy or not

redirectPortRewrite

The state of port rewrite feature support.

nonFipsCiphers

The state of usage of non FIPS approved ciphers.

ssl2

The state of SSLv2 protocol support.

ssl3

The state of SSLv3 protocol support.

tls1

The state of TLSv1.0 protocol support.

tls11

The state of TLSv1.1 protocol support.

tls12

The state of TLSv1.2 protocol support.

SNIEnable

The state of SNI extension. Server Name Indication (SNI) helps to enable SSL encryption on multiple subdomains if the domains are controlled by the same organization and share the same second-level domain name. State of SNI feature on service

cipherAliasName/cipherName/cipherGroupName

The name of the cipher group/alias/individual cipher bindings.

cipherName

The cipher group/alias/individual cipher configuration

description

The cipher suite description.

service

Service

certkeyName

The name of the certificate key pair binding.

policyName

The name of the SSL policy binding.

invoke

Invoke flag. This attribute is relevant only for ADVANCED policies

labelType

Type of policy label invocation.

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

serviceName

Service name.

ocspCheck

The state of the OCSP check parameter. (Mandatory/Optional)

pushEncTrigger

PUSH packet triggering encryption: Always, Ignore, Merge

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

CA

CA certificate.

SNICert

The name of the CertKey. Use this option to bind Certkey(s) which will be used in SNI processing.

eccCurveName

Named ECC curve bound to vserver/service.

stateflag

skipCAName

The flag is used to indicate whether this particular CA certificate's CA_Name needs to be sent to the SSL client while requesting for client certificate in a SSL handshake

sendCloseNotify

Enable sending SSL Close-Notify at the end of a transaction

dtlsProfileName

Name of the DTLS profile whose settings are to be applied to the virtual server.

dtlsFlag

The flag is used to indicate whether DTLS is set or not

sslProfile

SSL profile associated to vserver

devno

count

Example

An example of the output of the `show vserver sslvip` command is as follows: `sh ssl vserver`

ssl wrapkey

The following operations can be performed on "ssl wrapkey":

[create](#) | [rm](#) | [show](#)

create ssl wrapkey

Generates a wrap key.

Synopsis

```
create ssl wrapkey <wrapKeyName> {-password } {-salt }
```

Arguments

wrapKeyName

Name for the wrap key. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the wrap key is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my key" or 'my key').

password

Password string for the wrap key.

salt

Salt string for the wrap key.

Example

```
create wrapkey wrap1 -password wrapkey123 -salt wrapsalt123
```

rm ssl wrapkey

Removes all the wrap keys, or the specified wrap key, from the appliance.

Synopsis

```
rm ssl wrapkey <wrapKeyName> ...
```

Arguments

wrapKeyName

Name of the wrap key to remove.

Example

```
rm wrapkey wrap1
```

show ssl wrapkey

Display the wrap keys.

Synopsis

show ssl wrapkey

Outputs

wrapKeyName

Wrap key name.

devno

count

stateflag

Example

An example of output of 'show wrapkey' command is as shown below: sh wrapkey 1 WRAP key:

Stream Commands

The entities on which you can perform NetScaler CLI operations:

- [stream identifier](#)
- [stream selector](#)
- [stream session](#)

stream identifier

The following operations can be performed on "stream identifier":

add | **set** | **unset** | **rm** | **show** | **stat**

add stream identifier

Creates a stream identifier. A stream identifier specifies how data is collected and stored for an Action Analytics configuration.

Synopsis

```
add stream identifier <name> <selectorName> [-interval <positive_integer>] [-SampleCount <positive_integer>] [-sort <sort>]
```

Arguments

name

The name of stream identifier.

selectorName

Name of the selector to use with the stream identifier.

interval

Number of minutes of data to use when calculating session statistics (number of requests, bandwidth, and response times). The interval is a moving window that keeps the most recently collected data. Older data is discarded at regular intervals.

Default value: 1

Minimum value: 1

SampleCount

Size of the sample from which to select a request for evaluation. The smaller the sample count, the more accurate is the statistical data. To evaluate all requests, set the sample count to 1. However, such a low setting can result in excessive consumption of memory and processing resources.

Default value: 1

Minimum value: 1

Maximum value: 65535

sort

Sort stored records by the specified statistics column, in descending order. Performed during data collection, the sorting enables real-time data evaluation through NetScaler policies (for example, compression and caching policies) that use functions such as IS_TOP(n).

Possible values: REQUESTS, CONNECTIONS, RESPTIME, BANDWIDTH, NONE

Default value: REQUESTS

Example

```
add stream identifier stream_id top_url -interval 10 -sampleCount 1 -sort REQUESTS
```

set stream identifier

Modifies the specified parameters of a stream identifier. Parameters for which a default value is available revert to their default values.

Synopsys

set stream identifier <name> [-selectorName <string>] [-interval <positive_integer>] [-SampleCount <positive_integer>] [-sort <sort>]

Arguments

name

The name of stream identifier.

selectorName

Name of the selector to use with the stream identifier.

interval

Number of minutes of data to use when calculating session statistics (number of requests, bandwidth, and response times). The interval is a moving window that keeps the most recently collected data. Older data is discarded at regular intervals.

Default value: 1

Minimum value: 1

SampleCount

Size of the sample from which to select a request for evaluation. The smaller the sample count, the more accurate is the statistical data. To evaluate all requests, set the sample count to 1. However, such a low setting can result in excessive consumption of memory and processing resources.

Default value: 1

Minimum value: 1

Maximum value: 65535

sort

Sort stored records by the specified statistics column, in descending order. Performed during data collection, the sorting enables real-time data evaluation through NetScaler policies (for example, compression and caching policies) that use functions such as IS_TOP(n).

Possible values: REQUESTS, CONNECTIONS, RESPTIME, BANDWIDTH, NONE

Default value: REQUESTS

Example

```
set stream identifier stream_id -selectorName top_clients -interval 1 -sampleCount 1 -sort
```

unset stream identifier

Use this command to remove stream identifier settings. Refer to the set stream identifier command for meanings of the arguments.

Synopsys

unset stream identifier <name> [-selectorName] [-interval] [-SampleCount] [-sort]

rm stream identifier

Removes a stream identifier. Note: You cannot remove a stream identifier if it is being used in a policy.

Synopsys

rm stream identifier <name>

Arguments

name

The name of stream identifier.

Example

```
rm stream identifier stream_id
```

show stream identifier

Displays the parameters of the specified stream identifier or, if no stream identifier name is specified, the parameters of all configured stream identifiers.

Synopsys

show stream identifier [<name>]

Arguments

name

The name of stream identifier.

Outputs

selectorName

Name of the selector to use with the stream identifier.

rule

Rule.

ngname

Nodegroup name to which this identifier belongs to.

recordlimit

Maximum number of objects allowed per identifier.

interval

Number of minutes of data to use when calculating session statistics (number of requests, bandwidth, and response times). The interval is a moving window that keeps the most recently collected data. Older data is discarded at regular intervals.

SampleCount

Size of the sample from which to select a request for evaluation. The smaller the sample count, the more accurate is the statistical data. To evaluate all requests, set the sample count to 1. However, such a low setting can result in excessive consumption of memory and processing resources.

sort

Sort stored records by the specified statistics column, in descending order. Performed during data collection, the sorting enables real-time data evaluation through NetScaler policies (for example, compression and caching policies) that use functions such as IS_TOP(n).

log

Location where objects collected on the identifier will be logged.

logInterval

Time interval in minutes for logging the collected objects. Log interval should be greater than or equal to the interval of the stream identifier.

logLimit

Maximum number of objects to be logged in the log interval.

builtin

Flag to determine if stream identifier is built-in or not

stateflag

used internally to identify ip addresses returned.

devno

count

Example

```
show stream identifier stream_id
```

stat stream identifier

Displays the statistics that the NetScaler appliance has collected for the specified stream identifier.

Synopsys

```
stat stream identifier <name> [<pattern> ...] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile  
<input_filename>] [-clearstats ( basic | full )] [-sortBy <sortBy> [<sortOrder>]]
```

Arguments

name

Name of the stream identifier.

pattern

Values on which grouping is performed are displayed in the output as row titles. If grouping is performed on two or more fields, their values are separated by a question mark in the row title.

For example, consider a selector that contains the expressions HTTP.REQ.URL and CLIENT.IP.SRC (in that order), on an appliance that has accumulated records of a number of requests for two URLs, example.com/page1.html and example.com/page2.html, from two client IP addresses, 192.0.2.10 and 192.0.2.11.

With a pattern of ? ?, the appliance performs grouping on both fields and displays statistics for the following:

- * Requests for example.com/abc.html from 192.0.2.10, with a row title of example.com/abc.html?192.0.2.10.
- * Requests for example.com/abc.html from 192.0.2.11, with a row title of example.com/abc.html?192.0.2.11.
- * Requests for example.com/def.html from 192.0.2.10, with a row title of example.com/def.html?192.0.2.10.
- * Requests for example.com/def.html from 192.0.2.11, with a row title of example.com/def.html?192.0.2.11.

With a pattern of * ?, the appliance performs grouping on only the client IP address values and displays statistics for the following requests:

- * All requests from 192.0.2.10, with the IP address as the row title.
- * All requests from 192.0.2.11, with the IP address as the row title.

With a pattern of ? *, the appliance performs grouping on only the URL values and displays statistics for the following requests:

* All requests for example.com/abc.html, with the URL as the row title.

* All requests for example.com/def.html, with the URL as the row title.

With a pattern of * *, the appliance displays one set of collective statistics for all the requests received, with no row title.

With a pattern of example.com/abc.html ?, the appliance displays statistics for requests for example.com/abc.html from each unique client IP address.

With a pattern of * 192.0.2.11, the appliance displays statistics for all requests from 192.0.2.11.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

sortBy

use this argument to sort by specific key

Possible values: Req, BandW, RspTime, Conn

sortOrder

use this argument to specify sort order

Possible values: ascending, descending

Default value: SORT_DESCENDING

Outputs

count

devno

stateflag

Outputs

Stream Session Requests (Req)

Total number of Stream Requests recieved.

Stream Session Bandwidth (BandW)

Total Bandwidth consumed.

Stream Session Response Time (RspTime)

Average response time of the stream session.

Stream Session Connections (Conn)

Current connections on the stream session.

stream selector

The following operations can be performed on "stream selector":

[add](#) | [set](#) | [rm](#) | [show](#)

add stream selector

Creates a selector for Action Analytics or traffic rate limiting.

Synopsys

```
add stream selector <name> <rule> ...
```

Arguments

name

Name for the selector. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. If the name includes one or more spaces, and you are using the NetScaler CLI, enclose the name in double or single quotation marks (for example, "my selector" or 'my selector').

rule

Set of up to five individual (not compound) default syntax expressions. Maximum length: 7499 characters. Each expression must identify a specific request characteristic, such as the client's IP address (with CLIENT.IP.SRC) or requested server resource (with HTTP.REQ.URL).

Note: If two or more selectors contain the same expressions in different order, a separate set of records is created for each selector.

Example

```
add stream selector sel_subnet HTTP.REQ.URL CLIENT.IP.SRC.SUBNET(24)
```

set stream selector

Modifies the set of expressions in a stream selector. Note: You can change an expression if the selector is not yet being used in an identifier. If the selector is already in use, you can change only the order of the expressions, not the expressions themselves.

Synopsys

```
set stream selector <name> -rule <expression> ...
```

Arguments

name

Name of the selector for which to modify parameters.

rule

Set of up to five individual (not compound) default syntax expressions. Maximum length: 7499 characters. Each expression must identify a specific request characteristic, such as the client's IP address (with CLIENT.IP.SRC) or requested server resource (with HTTP.REQ.URL).

Note: If two or more selectors contain the same expressions in different order, a separate set of records is created for each selector.

Example

```
set stream sel_subnet HTTP.REQ.URL CLIENT.IP.SRC
```

rm stream selector

Removes a selector. Note: Before you remove a selector, make sure that it is not being used by an identifier.

Synopsys

```
rm stream selector <name>
```

Arguments

name

Name for the selector. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. If the name includes one or more spaces, and you are using the NetScaler CLI, enclose the name in double or single quotation marks (for example, "my selector" or 'my selector').

Example

```
rm stream selector sel_subnet
```

show stream selector

Displays the expressions configured for the specified selector or, if no selector name is specified, the expressions configured for all selectors.

Synopsys

```
show stream selector [<name>]
```

Arguments

name

Name for the selector. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. If the name includes one or more spaces, and you are using the NetScaler CLI, enclose the name in double or single quotation marks (for example, "my selector" or 'my selector').

Outputs

rule

Set of up to five individual (not compound) default syntax expressions. Maximum length: 7499 characters. Each expression must identify a specific request characteristic, such as the client's IP address (with CLIENT.IP.SRC) or requested server resource (with HTTP.REQ.URL).

Note: If two or more selectors contain the same expressions in different order, a separate set of records is created for each selector.

builtin

Flag to determine if stream selector is built-in or not

stateflag

devno

count

Example

```
show ns limitSelector sel_subnet
```

stream session

The following operations can be performed on "stream session":

clear stream session

Flushes all the records that have been accumulated for the specified stream identifier.

Synopsys

```
clear stream session <name>
```

Arguments

name

Name of the stream identifier.

Example

```
clear stream session stream_id
```

System Commands

The entities on which you can perform NetScaler CLI operations:

- o system
- o system backup
- o system bw
- o system cmdPolicy
- o system collectionparam
- o system core
- o system countergroup
- o system counters
- o system cpu
- o system dataSource
- o system entity
- o system entitydata
- o system entitytype
- o system eventhistory
- o system global
- o system globaldata
- o system group
- o system memory
- o system parameter
- o system session
- o system user

system

The following operations can be performed on "system":

stat system

This command displays system statistics

Synopsys

```
stat system [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

Maximum Memory Available (MB) (MemTotAvail)

Total system memory available for PE to grab from the system.

-12.0 V Supply Voltage (V12n)

Power supply -12V output. Acceptable range is -13.20 through -10.80 volts. 9800 and 9960 platforms display standard value of -12.0V.

-5.0 V Supply Voltage (V50n)

Power supply -5V output. Acceptable range is -5.50 through -4.50 volts. 9800 and 9960 platforms display standard value of -5.0V.

CPU usage (CPU)

CPU utilization: percentage * 10.

Average CPU usage (CPU)

Shows average CPU utilization percentage if more than 1 CPU is present.

CPU1

CPU 1 (currently the slave CPU) utilization, as percentage of capacity. Not applicable for a single-CPU system.

CPU0

CPU 0 (currently the master CPU) utilization, as percentage of capacity.

Voltage 7 (Volts) (volt7)

Voltage of a device connected to health monitoring chip through pin 7.

Voltage 6 (Volts) (volt6)

Voltage of a device connected to health monitoring chip through pin 6.

Voltage 5 (Volts) (volt5)

Voltage of a device connected to health monitoring chip through pin 5.

Voltage 4 (Volts) (volt4)

Voltage of a device connected to health monitoring chip through pin 4.

Voltage 3 (Volts) (volt3)

Voltage of a device connected to health monitoring chip through pin 3.

Voltage 2 (Volts) (volt2)

Voltage of a device connected to health monitoring chip through pin 2.

Voltage 1 (Volts) (volt1)

Voltage of a device connected to health monitoring chip through pin 1.

Voltage 0 (Volts) (volt0)

Voltage of a device connected to health monitoring chip through pin 0.

Voltage Sensor2(Volts) (VSEN2)

Voltage Sensor 2 Input. Currently only 13k Platforms will have valid value for this counter and for older platforms this will be 0.

5V Standby Voltage(Volts) (V5SB)

Power Supply 5V Standby Voltage. Currently only 13k Platforms will have valid value for this counter and for older platforms this will be 0.

Intel CPU Vtt Power(Volts) (VTT)

Intel CPU Vtt power. Currently only 13k Platforms will have valid value for this counter and for older platforms this will be 0.

Battery Voltage (Volts) (VBAT)

Onboard battery power supply output. 9800 and 9950 platforms display standard value of 5.0V.

+12.0 V Supply Voltage (V+12)

Power supply +12V output. Acceptable range is 10.80 through 13.20 volts.

+5.0 V Supply Voltage (V50)

Power supply +5V output. Acceptable range is 4.50 through 5.50 volts.

Standby 3.3 V Supply Voltage (V33Stby)

Standby power supply +3.3V output. Acceptable range is 2.970 through 3.630 volts. 9800 and 9960 platforms display standard value of 3.3V.

You can configure Standby 3.3V Supply Voltage by using the Set snmp alarm VOLTAGE-LOW command to set the lower limit and the Set snmp alarm VOLTAGE-HIGH command to set the upper limit.

Main 3.3 V Supply Voltage (V33Main)

Main power supply +3.3V output. Acceptable range is 2.970 through 3.630 volts. This is a critical counter.

You can configure Main 3.3V Supply Voltage, by using the Set snmp alarm VOLTAGE-LOW command to set the lower limit and the Set snmp alarm VOLTAGE-HIGH command to set the upper limit.

CPU 1 Core Voltage (Volts) (VCC1)

CPU core 1 voltage. Acceptable range is 1.080 through 1.650 volts. If CPU 1 is not connected to the health monitoring chip, display shows voltage of CPU 0.

CPU 0 Core Voltage (Volts) (VCC0)

CPU core 0 voltage. Acceptable range is 1.080 through 1.650 volts.

Number of CPUs (CPUs)

The number of CPUs on the NetScaler appliance.

InUse Memory (%) (MemUsage)

Percentage of memory utilization on NetScaler.

Memory usage (MB) (MemUseMB)

Main memory currently in use, in megabytes.

Management CPU usage (%) (CPU)

Management CPU utilization percentage.

Packet CPU usage (%) (CPU)

Average CPU utilization percentage for all packet engines excluding management PE.

CPU usage (%) (CPU)

CPU utilization percentage.

Average CPU usage (%) (CPU)

Average CPU utilization percentage. Not applicable for a single-CPU system.

Up since (Since)

Time when the NetScaler appliance was last started.

/flash Used (%) (disk0PerUsage)

Used space in /flash partition of the disk, as a percentage. This is a critical counter.

You can configure /flash Used (%) by using the Set snmp alarm DISK-USAGE-HIGH command.

/var Used (%) (disk1PerUsage)

Used space in /var partition of the disk, as a percentage. This is a critical counter. You can configure /var Used (%) by using the Set snmp alarm DISK-USAGE-HIGH command.

CPU Fan 0 Speed (RPM) (CPUFan0)

CPU Fan 0 speed. Acceptable range is 3000 through 6000 RPM. This is a critical counter.

You can configure CPU Fan 0 Speed by using the Set snmp alarm FAN-SPEED-LOW command to set the lower limit.

CPU Fan 1 Speed (RPM) (CPUFan1)

CPU Fan 1 speed. Acceptable range is 3000 through 6000 RPM. 7000 platform displays speed of CPU fan 0. This is a critical counter.

You can configure CPU Fan 1 Speed by using the Set snmp alarm FAN-SPEED-LOW command to set the lower limit.

System Fan Speed (RPM) (systemFan)

System fan speed. Acceptable range is 3000 through 6000 RPM. This is a critical counter.

You can configure System Fan Speed by using the Set snmp alarm FAN-SPEED-LOW command to set the lower limit.

System Fan 1 Speed (RPM) (systemFan1)

System fan 1 speed. For new platforms associated pin is connected to CPU supporting fans. For platforms in which it is not connected, it will point to System Fan.

System Fan 2 Speed (RPM) (systemFan2)

System fan 2 speed. For new platforms associated pin is connected to CPU supporting fans. For platforms in which it is not connected, it will point to System Fan

CPU 0 Temperature (Celsius) (TCPU0)

CPU 0 temperature. 9800 and 9960 platforms display internal chip temperature. This is a critical counter.

You can configure CPU 0 Temperature by using the Set snmp alarm TEMPERATURE-HIGH command to set the upper limit.

CPU 1 Temperature (Celsius) (TCPU1)

CPU 1 temperature. 9800 and 9960 platforms display internal chip temperature. 7000, 9010 and 10010 platforms display CPU 0 temperature. This is a critical counter.

You can configure CPU 1 Temperature by using the Set snmp alarm TEMPERATURE-HIGH command to set the upper limit.

Internal Temperature (Celsius) (intTemp)

Internal temperature of health monitoring chip. This is a critical counter.

You can configure Internal Temperature by using the Set snmp alarm TEMPERATURE-HIGH command to set the upper limit.

Power supply 1 status (PS1FAIL)

Power supply 1 failure status.

Power supply 2 status (PS2FAIL)

Power supply 2 failure status.

Power supply 3 status (PS3FAIL)

Power supply 3 failure status.

Power supply 4 status (PS4FAIL)

Power supply 4 failure status.

/flash Size (MB) (disk0Size)

Size of /flash partition of the hard disk.

/flash Used (MB) (disk0Used)

Used space in /flash partition of the hard disk.

/flash Available (MB) (disk0Avail)

Available space in /flash partition of the hard disk.

/var Size (MB) (disk1Size)

Size of /var partition of the hard disk.

/var Used (MB) (disk1Used)

Used space in /var partition of the hard disk.

/var Available (MB) (disk1Avail)

Available space in /var partition of the hard disk.

Fan 0 Speed (RPM) (Fan0)

Speed of Fan 0 if associated pin is connected to health monitoring chip.

Fan 1 Speed (RPM) (Fan1)

Speed of Fan 1 if associated pin is connected to health monitoring chip.

Fan 2 Speed (RPM) (Fan2)

Speed of Fan 2 if associated pin is connected to health monitoring chip.

Fan 3 Speed (RPM) (Fan3)

Speed of Fan 3 if associated pin is connected to health monitoring chip.

Temperature 0 (Celsius) (temp0)

Temperature of a device connected to health monitoring chip through pin 0.

Temperature 1 (Celsius) (temp1)

Temperature of a device connected to health monitoring chip through pin 1.

Temperature 2 (Celsius) (temp2)

Temperature of a device connected to health monitoring chip through pin 2.

Temperature 3 (Celsius) (temp3)

Temperature of a device connected to health monitoring chip through pin 3.

Up time (UP)

Seconds since the NetScaler appliance started.

System memory (MB) (Memory)

Total amount of system memory, in megabytes.

Management CPU usage (CPU)

Management CPU utilization: percentage * 10.

Master CPU usage (CPU0)

CPU0 utilization: percentage * 10.

Slave CPU usage (CPU1)

CPU1 utilization, percentage * 10.

system backup

The following operations can be performed on "system backup":

[create](#) | [restore](#) | [rm](#) | [show](#)

create system backup

Creates a backup file (*.tgz) that is stored in the /var/ns_sys_backup/ directory. This file can be used to restore the appliance by using the "restore system backup" command.

Synopsys

```
create system backup [<fileName>] [-level ( basic | full )] [-comment <string>]
```

Arguments

fileName

Name of the backup file (*.tgz) to be restored.

level

Level of data to be backed up.

Possible values: basic, full

Default value: basic

comment

Comment specified at the time of creation of the backup file (*.tgz).

restore system backup

Restores an appliance by using the backup file (*.tgz) that was created by using the "create system backup" command.

Synopsys

```
restore system backup <fileName>
```

Arguments

fileName

Name of the backup file (*.tgz) to be restored.

rm system backup

Removes a backup file (*.tgz) that was created by using the "create system backup" command.

Synopsys

```
rm system backup <fileName>
```

Arguments

fileName

Name of the backup file (*.tgz) to be restored.

show system backup

Retrieves the backed up files that were created in the appliance.

Synopsys

show system backup [<fileName>]

Arguments

fileName

Name of the backup file(*.tgz) to be restored.

Outputs

level

Level of data to be backed up.

comment

Comment specified at the time of creation of the backup file(*.tgz).

size

Size of the backup file(*.tgz) in KB.

creationTime

Creation time of the backup file(*.tgz).

version

Build version of the backup file(*.tgz).

createdBy

Name of user who created the backup file(*.tgz).

IPAddress

Ip of Netscaler box where the backup file(*.tgz) was created.

devno

count

stateflag

system bw

The following operations can be performed on "system bw":

stat system bw

Displays BW statistics

Synopsys

```
stat system bw [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

Number of HTTP Requests Sent (httpClttotpoolinactive)

No of requests sent from BW client.

Number of HTTP responses received (httpClttotpooloutactive)

No of responses Received.

Http responses (200 OK) (httpSvr200OKresp)

Number of 200 Ok response sent from the BW appliance.

Http response error (404 NotFound) (httpSvr404ObjNotFound)

Number of 404 Not Found responses sent

No of stray packets received without request (httpClterrstray)

Number of stray packets received from server without HTTP request

0.5 ms (httpCltTTFPBandLWM)

Number of Responses Falling on LWM for TTFP.

= 1.5 ms (`httpClitTTFPBand0`)

Number of Responses Falling on Band-0 for TTFP.

= 2.5 ms (`httpClitTTFPBand1`)

Number of Responses Falling on Band-1 for TTFP.

= 3.5 ms (`httpClitTTFPBand2`)

Number of Responses Falling on Band-2 for TTFP.

= 4.5 ms (`httpClitTTFPBand3`)

Number of Responses Falling on Band-3 for TTFP.

= 5.5 ms (`httpClitTTFPBand4`)

Number of Responses Falling on Band-4 for TTFP.

= 6.5 ms (`httpClitTTFPBand5`)

Number of Responses Falling on Band-5 for TTFP.

= 7.5 ms (`httpClitTTFPBand6`)

Number of Responses Falling on Band-6 for TTFP.

= 8.5 ms (`httpClitTTFPBand7`)

Number of Responses Falling on Band-7 for TTFP.

RTT Value > 8.5 ms (`httpClitTTFPBandHWM`)

Number of Responses Falling on HWM for TTFP.

TTFP Peak RTT (`httpClitTTFPMax`)

Peak RTT observed for Time to First response packet.

0.5 ms (`httpClitTTLPBandLWM`)

Number of Responses Falling on LWM for TTLP.

= 1.5 ms (`httpClitTTLPBand0`)

Number of Responses Falling on Band-0 for TTLP.

= 2.5 ms (`httpClitTTLPBand1`)

Number of Responses Falling on Band-1 for TTLP.

= 3.5 ms (`httpClitTTLPBand2`)

Number of Responses Falling on Band-2 for TTLP.

= 4.5 ms (`httpClitTTLPBand3`)

Number of Responses Falling on Band-3 for TTLP.

= 5.5 ms (`httpClitTTLPBand4`)

Number of Responses Falling on Band-4 for TTLP.

= 6.5 ms (`httpClitTTLPBand5`)

Number of Responses Falling on Band-5 for TTLP.

= 7.5 ms (`httpClitTTLPBand6`)

Number of Responses Falling on Band-6 for TTLP.

= 8.5 ms (httpClfTTLPBand7)

Number of Responses Falling on Band-7 for TTLP.

RTT Value > 8.5 ms (httpClfTTLPBandHWM)

Number of Responses Falling on HWM for TTLP.

TTLP peak RTT (httpClfTTLPMax)

Peak RTT observed for Time to Last response packet.

system cmdPolicy

The following operations can be performed on "system cmdPolicy":

[add](#) | [rm](#) | [set](#) | [show](#)

add system cmdPolicy

Adds a command policy to the system. A command policy specifies the access rights of the system user. By default, the appliance already has the following policies defined: * operator * read-only * network * superuser

Synopsis

add system cmdPolicy <policyName> <action> <cmdSpec>

Arguments

policyName

Name for a command policy. Must begin with a letter, number, or the underscore (_) character, and must contain only alphanumeric, hyphen (-), period (.), hash (#), space (), at (@), equal (=), colon (:), and underscore characters. Cannot be changed after the policy is created.

CLI Users: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my policy" or 'my policy').

action

Action to perform when a request matches the policy.

Possible values: ALLOW, DENY

cmdSpec

Regular expression specifying the data that matches the policy.

rm system cmdPolicy

Removes a command policy from the appliance. Note: You cannot remove command policies that are bound to a system user.

Synopsis

rm system cmdPolicy <policyName>

Arguments

policyName

Name of the command policy to remove.

set system cmdPolicy

Modifies the specified attributes of an existing command policy.

Synopsis

set system cmdPolicy <policyName> <action> <cmdSpec>

Arguments

policyName

Name of the command policy to be modified.

action

Action to perform when a request matches the policy.

Possible values: ALLOW, DENY

cmdSpec

Regular expression specifying the data that matches the policy.

show system cmdPolicy

Displays information about all configured system command policies, or about the specified policy.

Synopsis

show system cmdPolicy [<policyName>]

Arguments

policyName

Name of the system command policy about which to display information.

Outputs

action

The policy action.

cmdSpec

The matching rule that the policy will utilize.

builtin**devno****count****stateflag**

system collectionparam

The following operations can be performed on "system collectionparam":

[set](#) | [unset](#) | [show](#)

set system collectionparam

Modifies a collection parameters for historical charting in nscollect.ini file.

Synopsys

```
set system collectionparam [-logLevel <string>] [-dataPath <string>]
```

Arguments

logLevel

specify the log level. Possible values CRITICAL,WARNING,INFO,DEBUG1,DEBUG2

dataPath

specify the data path to the database.

unset system collectionparam

Use this command to remove system collectionparam settings.Refer to the set system collectionparam command for meanings of the arguments.

Synopsys

```
unset system collectionparam [-logLevel] [-dataPath]
```

show system collectionparam

Displays collection parameters for historical charting present in nscollect.ini file.

Synopsys

```
show system collectionparam
```

Outputs

communityName

SNMPv1 community name for authentication.

logLevel

specify the log level. Possible values CRITICAL,WARNING,INFO,DEBUG1,DEBUG2

dataPath

specify the data path to the database.

system core

The following operations can be performed on "system core":

show system core

Display entities in historical data.

Synopsis

```
show system core [-dataSource <string>]
```

Arguments

dataSource

Specifies the source which contains all the stored counter values.

Outputs

response

system countergroup

The following operations can be performed on "system countergroup":

show system countergroup

Display available counter groups.

Synopsys

show system countergroup [-dataSource <string>]

Arguments

dataSource

Specifies the source which contains all the stored counter values.

Outputs

response

system counters

The following operations can be performed on "system counters":

show system counters

Display entities in historical data.

Synopsys

```
show system counters [<countergroup>] [-dataSource <string>]
```

Arguments

countergroup

Specify the (counter) group name which contains all the counters specific tot his particular group.

dataSource

Specifies the source which contains all the stored counter values.

Outputs

response

system cpu

The following operations can be performed on "system cpu":

stat system cpu

Displays statistics of all CPUs available on the appliance, or statistics of the specified CPU.

Synopsys

```
stat system cpu [<id>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (
basic | full )]
```

Arguments

id

ID of the CPU for which to display statistics.

Default value: 65535

Minimum value: 0

Maximum value: 65534

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

CPU Usage (Usage)

CPU utilization percentage.

system dataSource

The following operations can be performed on "system dataSource":

show system dataSource

Display entities in historical data.

Synopsys

show system dataSource [<dataSource>]

Arguments

dataSource

Specifies the source which contains all the stored counter values.

Outputs

response

system entity

The following operations can be performed on "system entity":

show system entity

Display entities in historical data.

Synopsys

```
show system entity <type> [-dataSource <string>] [-core <integer>]
```

Arguments

type

Specify the entity type.

dataSource

Specifies the source which contains all the stored counter values.

core

Specify core ID of the PE in nCore.

Outputs

response

Example

```
show system entity lbvserver
```

system entitydata

The following operations can be performed on "system entitydata":

[rm](#) | [show](#)

rm system entitydata

Removes the specified entity from historical charting along with all the associated counters till the current time stamp.

Synopsys

```
rm system entitydata [<type>] [<name>] [-allDeleted] [-allInactive] [-dataSource <string>] [-core <integer>]
```

Arguments

type

Specify the entity type.

name

Specify the entity name.

allDeleted

Specify this if you would like to delete information about all deleted entities from the database.

allInactive

Specify this if you would like to delete information about all inactive entities from the database.

dataSource

Specifies the source which contains all the stored counter values.

core

Specify core ID of the PE in nCore.

show system entitydata

Display the historical data for entity specific counters.

Synopsys

```
show system entitydata <type> <name> <counters> [-startTime <string> | (-last <integer> [<unit>])] [-endTime <string>] [-dataSource <string>] [-core <integer>]
```

Arguments

type

Specify the entity type.

name

Specify the entity name.

counters

Specify the counters to be collected.

startTime

Specify start time in mmddyyyyhhmm to start collecting values from that timestamp.

endTime

Specify end time in mmddyyyyhhmm upto which values have to be collected.

last

Last is literal way of saying a certain time period from the current moment. Example: -last 1 hour, -last 1 day, et cetera.

Default value: 1

unit

Specify the time period from current moment. Example 1 x where x = hours/ days/ years.

Possible values: HOURS, DAYS, MONTHS

dataSource

Specifies the source which contains all the stored counter values.

core

Specify core ID of the PE in nCore.

Outputs

response

startUpdate

lastupdate

Example

```
show system entitydata lbvserver v1 totalrequests -last 1 days
```

system entitytype

The following operations can be performed on "system entitytype":

show system entitytype

Display available entity types.

Synopsys

```
show system entitytype [-dataSource <string>]
```

Arguments

dataSource

Specifies the source which contains all the stored counter values.

Outputs

response

system eventhistory

The following operations can be performed on "system eventhistory":

show system eventhistory

Display events in historical data.

Synopsys

```
show system eventhistory [-startTime <string> | (-last <integer> [<unit>])] [-endTime <string>] -dataSource <string>
```

Arguments

startTime

Specify start time in mmddyyyyhhmm to start collecting values from that timestamp.

endTime

Specify end time in mmddyyyyhhmm upto which values have to be collected.

last

Last is literal way of saying a certain time period from the current moment. Example: -last 1 hour, -last 1 day, et cetera.

Default value: 1

unit

Specify the time period from current moment. Example 1 x where x = hours/ days/ years.

Possible values: HOURS, DAYS, MONTHS

dataSource

Specifies the source which contains all the stored counter values.

Outputs

response

system global

The following operations can be performed on "system global":

[bind](#) | [unbind](#) | [show](#)

bind system global

Binds policies globally.

Synopsys

```
bind system global [<policyName> [-priority <positive_integer>]]
```

Arguments

policyName

Name of the policy to bind globally.

priority

Integer specifying the priority of the policy. A lower number specifies a higher priority. Policies are evaluated in the order of their priority numbers. Note that priority range 64001 to 65535 is reserved for internal system usage of binding policies by default

Minimum value: 0

Maximum value: 65535

unbind system global

Unbinds a globally bound policy.

Synopsys

```
unbind system global <policyName>
```

Arguments

policyName

Name of the globally bound policy to unbind.

show system global

Displays information about all global policy bindings.

Synopsys

```
show system global
```

Outputs

policyName

The name of the command policy.

priority

The priority of the command policy.

bindPolicyType

Bound policy type

policySubType

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

stateflag

devno

count

system globaldata

The following operations can be performed on "system globaldata":

show system globaldata

Display historical data for global counters.

Synopsys

```
show system globaldata <counters> [<countergroup>] [-startTime <string> | (-last <integer> [<unit>])] [-endTime  
<string>] [-dataSource <string>] [-core <integer>]
```

Arguments

counters

Specify the counters to be collected.

countergroup

Specify the (counter) group name which contains all the counters specific to this particular group.

startTime

Specify start time in mmddyyyyhhmm to start collecting values from that timestamp.

endTime

Specify end time in mmddyyyyhhmm upto which values have to be collected.

last

Last is literal way of saying a certain time period from the current moment. Example: -last 1 hour, -last 1 day, et cetera.

Default value: 1

unit

Specify the time period from current moment. Example 1 x where x = hours/ days/ years.

Possible values: HOURS, DAYS, MONTHS

dataSource

Specifies the source which contains all the stored counter values.

core

Specify core ID of the PE in nCore.

Outputs

response

startUpdate

lastupdate

Example

```
show system globaldata cpu_usage -last 1 hours
```

system group

The following operations can be performed on "system group":

add | **rm** | **bind** | **unbind** | **show** | **set** | **unset**

add system group

Creates a system-user group, to which you can bind individual users by using the bind system group command.

Synopsis

```
add system group <groupName> [-promptString <string>] [-timeout <secs>]
```

Arguments

groupName

Name for the group. Must begin with a letter, number, or the underscore (_) character, and must contain only alphanumeric, hyphen (-), period (.), hash (#), space (), at (@), equal (=), colon (:), and underscore characters. Cannot be changed after the group is created.

CLI Users: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my group" or 'my group').

promptString

String to display at the command-line prompt. Can consist of letters, numbers, hyphen (-), period (.), hash (#), space (), at (@), equal (=), colon (:), underscore (_), and the following variables:

- * %u - Will be replaced by the user name.
- * %h - Will be replaced by the hostname of the NetScaler appliance.
- * %t - Will be replaced by the current time in 12-hour format.
- * %T - Will be replaced by the current time in 24-hour format.
- * %d - Will be replaced by the current date.
- * %s - Will be replaced by the state of the NetScaler appliance.

Note: The 63-character limit for the length of the string does not apply to the characters that replace the variables.

timeout

CLI session inactivity timeout, in seconds. If Restrictedtimeout argument of system parameter is enabled, Timeout can have values in the range [300-86400] seconds. If Restrictedtimeout argument of system parameter is disabled, Timeout can have values in the range [0, 10-100000000] seconds. Default value is 900 seconds.

rm system group

Removes a system group from the appliance.

Synopsis

```
rm system group <groupName>
```

Arguments

groupName

Name of the system group to remove.

bind system group

Binds a system user to a system group.

Synopsis

```
bind system group <groupName> [-userName <string>] [-policyName <string> <priority>]
```

Arguments

groupName

Name of the system group.

userName

Name of a system user to bind to the group.

policyName

Name of the command policy to be bind to the group.

priority

Integer specifying the priority of the command policy. A lower number specifies a higher priority. Policies are evaluated in the order of their priority numbers.

Minimum value: 0

unbind system group

Unbinds a system user from a group.

Synopsis

```
unbind system group <groupName> [-userName <string>] [-policyName <string>]
```

Arguments

groupName

Name of the system group from which to unbind the user.

userName

Name of the system user to unbind from the group.

policyName

Command policy to unbind from the group.

show system group

Displays information about all system groups configured on the appliance, or about the specified group.

Synopsis

```
show system group [<groupName>]
```

Arguments

groupName

Name of the system group about which to display information.

Outputs

userName

The system user.

policyName

The name of command policy.

priority

The priority of the command policy.

promptString

String to display at the command-line prompt. Can consist of letters, numbers, hyphen (-), period (.), hash (#), space (), at (@), equal (=), colon (:), underscore (_), and the following variables:

- * %u - Will be replaced by the user name.
- * %h - Will be replaced by the hostname of the NetScaler appliance.
- * %t - Will be replaced by the current time in 12-hour format.
- * %T - Will be replaced by the current time in 24-hour format.
- * %d - Will be replaced by the current date.
- * %s - Will be replaced by the state of the NetScaler appliance.

Note: The 63-character limit for the length of the string does not apply to the characters that replace the variables.

timeout

CLI session inactivity timeout, in seconds. If Restrictedtimeout argument of system parameter is enabled, Timeout can have values in the range [300-86400] seconds. If Restrictedtimeout argument of system parameter is disabled, Timeout can have values in the range [0, 10-100000000] seconds. Default value is 900 seconds.

devno

count

stateflag

set system group

Modifies the specified parameters of a system group.

Synopsys

```
set system group <groupName> [-promptString <string>] [-timeout <secs>]
```

Arguments

groupName

Name of system group to be modified.

promptString

String to display at the command-line prompt. Can consist of letters, numbers, hyphen (-), period (.), hash (#), space (), at (@), equal (=), colon (:), underscore (_), and the following variables:

- * %u - Will be replaced by the user name.

* %h - Will be replaced by the hostname of the NetScaler appliance.

* %t - Will be replaced by the current time in 12-hour format.

* %T - Will be replaced by the current time in 24-hour format.

* %d - Will be replaced by the current date.

* %s - Will be replaced by the state of the NetScaler appliance.

Note: The 63-character limit for the length of the string does not apply to the characters that replace the variables.

timeout

CLI session inactivity timeout, in seconds. If Restrictedtimeout argument of system parameter is enabled, Timeout can have values in the range [300-86400] seconds. If Restrictedtimeout argument of system parameter is disabled, Timeout can have values in the range [0, 10-100000000] seconds. Default value is 900 seconds.

unset system group

Use this command to remove system group settings. Refer to the set system group command for meanings of the arguments.

Synopsys

unset system group <groupName> [-promptString] [-timeout]

system memory

The following operations can be performed on "system memory":

stat system memory

Displays system-memory statistics.

Synopsys

```
stat system memory [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

Shared Memory InUse (%) (shMemAllocpcnt)

Shared memory insue percent.

Shared Memory InUse (MB) (shMemAllocMB)

Shared memory insue, in megabytes.

Total Shared Memory (MB) (shMemtotMB)

Total shared memory allowed to allocate, in megabytes.

Free Memory (MB) (MemTotFree)

Total Free PE Memory in the System.

InUse Memory (%) (MemUsage)

Percentage of memory utilization on NetScaler.

InUse Memory (MB) (MemTotUseMB)

Total NetScaler Memory in use, in megabytes.

Memory Allocated (%) (MemTotAlloc(%))

Currently allocated memory in percent.

Memory Allocated (MB) (MemTotAlloc)

Currently allocated memory, in megabytes.

Memory Currently Available (MB) (MemTotMB)

Total memory available (grabbed) for use by packet engine (PE), in megabytes.

Maximum Memory Available (MB) (MemTotAvail)

Total system memory available for PE to grab from the system.

Example

```
stat system memory
```

system parameter

The following operations can be performed on "system parameter":

[set](#) | [unset](#) | [show](#)

set system parameter

Modifies the specified system parameters.

Synopsis

```
set system parameter [-rbaOnResponse ( ENABLED | DISABLED )] [-promptString <string>] [-natPcbForceFlushLimit <positive_integer>] [-natPcbRstOnTimeout ( ENABLED | DISABLED )] [-timeout <secs>] [-localAuth ( ENABLED | DISABLED )] [-restrictedtimeout ( ENABLED | DISABLED )]
```

Arguments

rbaOnResponse

Enable or disable Role-Based Authentication (RBA) on responses.

Possible values: ENABLED, DISABLED

Default value: ENABLED

promptString

String to display at the command-line prompt. Can consist of letters, numbers, hyphen (-), period (.), hash (#), space (), at (@), equal (=), colon (:), underscore (_), and the following variables:

- * %u - Will be replaced by the user name.
- * %h - Will be replaced by the hostname of the NetScaler appliance.
- * %t - Will be replaced by the current time in 12-hour format.
- * %T - Will be replaced by the current time in 24-hour format.
- * %d - Will be replaced by the current date.
- * %s - Will be replaced by the state of the NetScaler appliance.

Note: The 63-character limit for the length of the string does not apply to the characters that replace the variables.

natPcbForceFlushLimit

Flush the system if the number of Network Address Translation Protocol Control Blocks (NATPCBs) exceeds this value.

Default value: 2147483647

Minimum value: 1000

natPcbRstOnTimeout

Send a reset signal to client and server connections when their NATPCBs time out. Avoids the buildup of idle TCP connections on both the sides.

Possible values: ENABLED, DISABLED

Default value: DISABLED

timeout

CLI session inactivity timeout, in seconds. If Restrictedtimeout argument of system parameter is enabled, Timeout can have values in the range [300-86400] seconds. If Restrictedtimeout argument of system

parameter is disabled, Timeout can have values in the range [0, 10-100000000] seconds. Default value is 900 seconds.

localAuth

When enabled, local users can access NetScaler even when external authentication is configured. When disabled, local users are not allowed to access the NetScaler, Local users can access the NetScaler only when the configured external authentication servers are unavailable.

Possible values: ENABLED, DISABLED

Default value: ENABLED

restrictedtimeout

Enable/Disable the restricted timeout behaviour. When enabled, timeout cannot be configured beyond admin configured timeout and also it will have\

the [minimum - maximum] range check. When disabled, timeout will have the old behaviour. By default the value is disabled

Possible values: ENABLED, DISABLED

Default value: DISABLED

unset system parameter

Use this command to remove system parameter settings. Refer to the set system parameter command for meanings of the arguments.

Synopsis

unset system parameter [-rbaOnResponse] [-promptString] [-natPcbForceFlushLimit] [-natPcbRstOnTimeout] [-timeout] [-localAuth] [-restrictedtimeout]

show system parameter

Displays information about the system parameters.

Synopsis

show system parameter

Outputs

rbaOnResponse

Enable or disable Role-Based Authentication (RBA) on responses.

promptString

The global system prompt.

natPcbForceFlushLimit

Flush the system if the number of Network Address Translation Protocol Control Blocks (NATPCBs) exceeds this value.

natPcbRstOnTimeout

Send RST to client and server connections when the natpcbs timeout. This avoids the buildup of idle TCP connections on both sides.

timeout

CLI session inactivity timeout, in seconds. If Restrictedtimeout argument of system parameter is enabled, Timeout can have values in the range [300-86400] seconds. If Restrictedtimeout argument of system parameter is disabled, Timeout can have values in the range [0, 10-100000000] seconds. Default value is 900 seconds.

maxClient

Maximum number of client connection allowed by the system

localAuth

When enabled, local users can access NetScaler even when external authentication is configured. When disabled, local users are not allowed to access the NetScaler, Local users can access the NetScaler only when the configured external authentication servers are unavailable.

restrictedtimeout

Enable/Disable the restricted timeout behaviour. When enabled, timeout cannot be configured beyond admin configured timeout and also it will have\\

the [minimum - maximum] range check. When disabled, timeout will have the old behaviour. By default the value is disabled

system session

The following operations can be performed on "system session":

[show](#) | [kill](#)

show system session

Displays information about all current system sessions, or about the specified session. The system might reclaim sessions with no active connections before expiry time.

Synopsys

show system session [<sid>]

Arguments

sid

ID of the system session about which to display information.

Minimum value: 1

Outputs

userName

user name of the session

logintime

logged-in time of this session

lastactivitytime

last activity time of on this session

expirytime

Time left in expire the session in seconds

numOfconnections

number of connection using this token

currentconn

True if the token is used for current session

devno

count

stateflag

kill system session

Kills one system session, or all system sessions except the current session.

Synopsys

kill system session (<sid> | -all)

Arguments

sid

ID of the system session to terminate.

CLI users: You can get the session ID by using the show system session command.

Minimum value: 1

all

Terminate all the system sessions except the current session.

system user

The following operations can be performed on "system user":

add | **rm** | **set** | **unset** | **bind** | **unbind** | **show**

add system user

Adds a new user to the system. Note: You must provide the password after the user name.

Synopsys

```
add system user <userName> [-externalAuth ( ENABLED | DISABLED )] [-promptString <string>] [-timeout <secs>] [-logging ( ENABLED | DISABLED )]
```

Arguments

userName

Name for a user. Must begin with a letter, number, or the underscore (`_`) character, and must contain only alphanumeric, hyphen (`-`), period (`.`), hash (`#`), space (), at (`@`), equal (`=`), colon (`:`), and underscore characters. Cannot be changed after the user is added.

CLI Users: If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my user" or 'my user').

password

Password for the system user. Can include any ASCII character.

externalAuth

Whether to use external authentication servers for the system user authentication or not

Possible values: ENABLED, DISABLED

Default value: ENABLED

promptString

String to display at the command-line prompt. Can consist of letters, numbers, hyphen (`-`), period (`.`), hash (`#`), space (), at (`@`), equal (`=`), colon (`:`), underscore (`_`), and the following variables:

- * `%u` - Will be replaced by the user name.
- * `%h` - Will be replaced by the hostname of the NetScaler appliance.
- * `%t` - Will be replaced by the current time in 12-hour format.
- * `%T` - Will be replaced by the current time in 24-hour format.
- * `%d` - Will be replaced by the current date.
- * `%s` - Will be replaced by the state of the NetScaler appliance.

Note: The 63-character limit for the length of the string does not apply to the characters that replace the variables.

timeout

CLI session inactivity timeout, in seconds. If Restrictedtimeout argument of system parameter is enabled, Timeout can have values in the range [300-86400] seconds. If Restrictedtimeout argument of system parameter is disabled, Timeout can have values in the range [0, 10-100000000] seconds. Default value is 900 seconds.

logging

Users logging privilege

Possible values: ENABLED, DISABLED

Default value: DISABLED

rm system user

Removes a system user from the appliance.

Synopsis

```
rm system user <userName>
```

Arguments

userName

Name of the system user to remove.

set system user

Modifies the specified parameters of a system-user entry.

Synopsis

```
set system user <userName> {-password } [-externalAuth ( ENABLED | DISABLED )] [-promptString <string>] [-  
timeout <secs>] [-logging ( ENABLED | DISABLED )]
```

Arguments

userName

Name of the system-user entry to modify.

password

Password for the system user. Can include any ASCII character.

externalAuth

Whether to use external authentication servers for the system user authentication or not

Possible values: ENABLED, DISABLED

Default value: ENABLED

promptString

String to display at the command-line prompt. Can consist of letters, numbers, hyphen (-), period (.), hash (#), space (), at (@), equal (=), colon (:), underscore (_), and the following variables:

* %u - Will be replaced by the user name.

* %h - Will be replaced by the hostname of the NetScaler appliance.

* %t - Will be replaced by the current time in 12-hour format.

* %T - Will be replaced by the current time in 24-hour format.

* %d - Will be replaced by the current date.

* %s - Will be replaced by the state of the NetScaler appliance.

Note: The 63-character limit for the length of the string does not apply to the characters that replace the variables.

timeout

CLI session inactivity timeout, in seconds. If Restrictedtimeout argument of system parameter is enabled, Timeout can have values in the range [300-86400] seconds. If Restrictedtimeout argument of system parameter is disabled, Timeout can have values in the range [0, 10-100000000] seconds. Default value is 900 seconds.

logging

Users logging privilege

Possible values: ENABLED, DISABLED

Default value: DISABLED

unset system user

Use this command to remove system user settings. Refer to the set system user command for meanings of the arguments.

Synopsis

```
unset system user <userName> [-externalAuth] [-promptString] [-timeout] [-logging]
```

bind system user

Binds a command policy to a system user.

Synopsis

```
bind system user <userName> <policyName> <priority>
```

Arguments

userName

Name of the system-user entry to which to bind the command policy.

policyName

Name of the command policy to bind to the system user.

priority

Integer specifying the priority of the command policy. A lower number specifies a higher priority. Policies are evaluated in the order of their priority numbers.

Minimum value: 0

Maximum value: 999999999

unbind system user

Unbinds a command policy from the system user.

Synopsis

```
unbind system user <userName> <policyName>
```

Arguments

userName

Name of the user entry from which to unbind the command policy.

policyName

Name of the command policy to unbind.

show system user

Displays information about all system users configured on the appliance, or about the specified user.

Synopsys

show system user [<userName>]

Arguments

userName

Name of a system user about whom to display information.

Outputs

groupName

The system group.

policyName

The name of command policy.

priority

The priority of the policy.

password

Password for the system user. Can include any ASCII character.

encrypted

externalAuth

Whether to use external authentication servers for the system user authentication or not

promptString

String to display at the command-line prompt. Can consist of letters, numbers, hyphen (-), period (.), hash (#), space (), at (@), equal (=), colon (:), underscore (_), and the following variables:

- * %u - Will be replaced by the user name.
- * %h - Will be replaced by the hostname of the NetScaler appliance.
- * %t - Will be replaced by the current time in 12-hour format.
- * %T - Will be replaced by the current time in 24-hour format.
- * %d - Will be replaced by the current date.
- * %s - Will be replaced by the state of the NetScaler appliance.

Note: The 63-character limit for the length of the string does not apply to the characters that replace the variables.

promptInheritedFrom

From where the prompt has been inherited.

timeout

CLI session inactivity timeout, in seconds. If Restrictedtimeout argument of system parameter is enabled, Timeout can have values in the range [300-86400] seconds. If Restrictedtimeout argument of system parameter is disabled, Timeout can have values in the range [0, 10-100000000] seconds. Default value is 900 seconds.

timeoutKind

From where the timeout has been inherited.

logging

Users logging privilege

devno**count****stateflag**

Traffic Management Commands

The entities on which you can perform NetScaler CLI operations:

- o `tm formSSOAction`
- o `tm global`
- o `tm samlSSOProfile`
- o `tm sessionAction`
- o `tm sessionParameter`
- o `tm sessionPolicy`
- o `tm trafficAction`
- o `tm trafficPolicy`

tm formSSOAction

The following operations can be performed on "tm formSSOAction":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add tm formSSOAction

Creates a form-based single sign-on traffic profile (action.) Form-based single sign-on allows users to access web applications that require an HTML form-based logon without having to type their password again for each new application.

Synopsys

```
add tm formSSOAction <name> -actionURL <URL> -userField <string> -passwdField <string> -ssoSuccessRule
<expression> [-nameValuePair <string>] [-responsesize <positive_integer>] [-nvtype ( STATIC | DYNAMIC )] [-
submitMethod ( GET | POST )]
```

Arguments

name

Name for the new form-based single sign-on profile. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after an SSO action is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my action" or 'my action').

actionURL

URL to which the completed form is submitted.

userField

Name of the form field in which the user types in the user ID.

passwdField

Name of the form field in which the user types in the password.

ssoSuccessRule

Expression, that checks to see if single sign-on is successful.

nameValuePair

Name-value pair attributes to send to the server in addition to sending the username and password. Value names are separated by an ampersand (&) (for example, name1=value1&name2=value2).

responsesize

Number of bytes, in the response, to parse for extracting the forms.

Default value: 8096

Minimum value: 0

nvtype

Type of processing of the name-value pair. If you specify STATIC, the values configured by the administrator are used. For DYNAMIC, the response is parsed, and the form is extracted and then submitted.

Possible values: STATIC, DYNAMIC

Default value: DYNAMIC

submitMethod

HTTP method used by the single sign-on form to send the logon credentials to the logon server. Applies only to STATIC name-value type.

Possible values: GET, POST

Default value: GET

rm tm formSSOAction

Deletes an existing form-based single sign-on traffic profile (action.)

Synopsis

```
rm tm formSSOAction <name>
```

Arguments

name

Name of the form-based single sign-on profile to delete.

set tm formSSOAction

Modifies the specified attributes of a form-based single sign-on traffic profile (action.)

Synopsis

```
set tm formSSOAction <name> [-actionURL <URL>] [-userField <string>] [-passwdField <string>] [-ssoSuccessRule <expression>] [-responsesize <positive_integer>] [-nameValuePair <string>] [-nvttype ( STATIC | DYNAMIC )] [-submitMethod ( GET | POST )]
```

Arguments

name

Name of the form-based single sign-on profile (action) to modify.

actionURL

URL to which the completed form is submitted.

userField

Name of the form field in which the user types in the user ID.

passwdField

Name of the form field in which the user types in the password.

ssoSuccessRule

Expression, that checks to see if single sign-on is successful.

responsesize

Number of bytes, in the response, to parse for extracting the forms.

Default value: 8096

Minimum value: 0

nameValuePair

Name-value pair attributes to send to the server in addition to sending the username and password. Value names are separated by an ampersand (&) (for example, name1=value1&name2=value2).

nvtype

Type of processing of the name-value pair. If you specify STATIC, the values configured by the administrator are used. For DYNAMIC, the response is parsed, and the form is extracted and then submitted.

Possible values: STATIC, DYNAMIC

Default value: DYNAMIC

submitMethod

HTTP method used by the single sign-on form to send the logon credentials to the logon server. Applies only to STATIC name-value type.

Possible values: GET, POST

Default value: GET

unset tm formSSOAction

Use this command to remove tm formSSOAction settings. Refer to the set tm formSSOAction command for meanings of the arguments.

Synopsys

unset tm formSSOAction <name> [-responsesize] [-nameValuePair] [-nvtype] [-submitMethod]

show tm formSSOAction

Displays information about all configured form-based single sign-on actions, or displays detailed information about the specified action.

Synopsys

show tm formSSOAction [<name>]

Arguments

name

Name of the SSO action for which to display detailed information.

Outputs

actionURL

URL to which the completed form is submitted.

userField

Username field.

passwdField

Password field.

responsesize

Number of bytes, in the response, to parse for extracting the forms.

nameValuePair

Form attributes and their values to be submitted.

nvtype

Bypass Form extraction

ssoSuccessRule

Rule to be evaluated to check whether sso succeeded or not.

submitMethod

Form Submit method.

devno

count

stateflag

tm global

The following operations can be performed on "tm global":

[bind](#) | [unbind](#) | [show](#)

bind tm global

Binds traffic, sessions, nslog, and syslog policies to traffic management (TM) Global.

Synopsys

```
bind tm global [-policyName <string> [-priority <positive_integer>]]
```

Arguments

policyName

Name of the policy that you are binding.

priority

Integer specifying the policy's priority. The lower the number, the higher the priority. Policies are evaluated in the order of their priority numbers.

Minimum value: 0

unbind tm global

Unbinds a globally bound traffic session policy.

Synopsys

```
unbind tm global -policyName <string>
```

Arguments

policyName

Name of the policy to unbind.

show tm global

Displays information about TM global bindings.

Synopsys

```
show tm global
```

Outputs

policyName

The name of the policy.

priority

The priority of the policy.

type

Bindpoint to which the policy is bound

policySubType

stateflag

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

bindPolicyType

Bound policy type

devno

count

tm samlSSOProfile

The following operations can be performed on "tm samlSSOProfile":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add tm samlSSOProfile

Creates a SAML single sign-on profile. This profile is employed in triggering saml assertion to a target service based on traffic profile.

Synopsys

```
add tm samlSSOProfile <name> [-samlSigningCertName <string>] -assertionConsumerServiceURL <URL> -
relaystateRule <expression> [-sendPassword ( ON | OFF )] [-samlIssuerName <string>] [-signatureAlg ( RSA-SHA1
| RSA-SHA256 )] [-digestMethod ( SHA1 | SHA256 )] [-audience <string>] [-NameIDFormat <NameIDFormat>] [-
NameIDExpr <string>] [-Attribute1 <string> -Attribute1Expr <string> [-Attribute1FriendlyName <string>] [-
Attribute1Format ( URI | Basic )]] [-Attribute2 <string> -Attribute2Expr <string> [-Attribute2FriendlyName <string>] [-
Attribute2Format ( URI | Basic )]] [-Attribute3 <string> -Attribute3Expr <string> [-Attribute3FriendlyName <string>] [-
Attribute3Format ( URI | Basic )]] [-Attribute4 <string> -Attribute4Expr <string> [-Attribute4FriendlyName <string>] [-
Attribute4Format ( URI | Basic )]] [-Attribute5 <string> -Attribute5Expr <string> [-Attribute5FriendlyName <string>] [-
Attribute5Format ( URI | Basic )]] [-Attribute6 <string> -Attribute6Expr <string> [-Attribute6FriendlyName <string>] [-
Attribute6Format ( URI | Basic )]] [-Attribute7 <string> -Attribute7Expr <string> [-Attribute7FriendlyName <string>] [-
Attribute7Format ( URI | Basic )]] [-Attribute8 <string> -Attribute8Expr <string> [-Attribute8FriendlyName <string>] [-
Attribute8Format ( URI | Basic )]] [-Attribute9 <string> -Attribute9Expr <string> [-Attribute9FriendlyName <string>] [-
Attribute9Format ( URI | Basic )]] [-Attribute10 <string> -Attribute10Expr <string> [-Attribute10FriendlyName
<string>] [-Attribute10Format ( URI | Basic )]] [-Attribute11 <string> -Attribute11Expr <string> [-
Attribute11FriendlyName <string>] [-Attribute11Format ( URI | Basic )]] [-Attribute12 <string> -Attribute12Expr
<string> [-Attribute12FriendlyName <string>] [-Attribute12Format ( URI | Basic )]] [-Attribute13 <string> -
Attribute13Expr <string> [-Attribute13FriendlyName <string>] [-Attribute13Format ( URI | Basic )]] [-Attribute14
<string> -Attribute14Expr <string> [-Attribute14FriendlyName <string>] [-Attribute14Format ( URI | Basic )]] [-
Attribute15 <string> -Attribute15Expr <string> [-Attribute15FriendlyName <string>] [-Attribute15Format ( URI | Basic
)]] [-Attribute16 <string> -Attribute16Expr <string> [-Attribute16FriendlyName <string>] [-Attribute16Format ( URI |
Basic )]]
```

Arguments

name

Name for the new saml single sign-on profile. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after an SSO action is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my action" or 'my action').

samlSigningCertName

Name of the signing authority as given in the SAML server's SSL certificate.

assertionConsumerServiceURL

URL to which the assertion is to be sent.

relaystateRule

Expression to extract relaystate to be sent along with assertion. Evaluation of this expression should return TEXT content. This is typically a targ

et url to which user is redirected after the recipient validates SAML token

sendPassword

Option to send password in assertion.

Possible values: ON, OFF

Default value: OFF

samlIssuerName

The name to be used in requests sent from Netscaler to IdP to uniquely identify Netscaler.

signatureAlg

Algorithm to be used to sign/verify SAML transactions

Possible values: RSA-SHA1, RSA-SHA256

Default value: RSA-SHA1

digestMethod

Algorithm to be used to compute/verify digest for SAML transactions

Possible values: SHA1, SHA256

Default value: SHA1

audience

Audience for which assertion sent by IdP is applicable. This is typically entity name or url that represents ServiceProvider

Maximum value: 256

NameIDFormat

Format of Name Identifier sent in Assertion.

Possible values: Unspecified, emailAddress, X509SubjectName, WindowsDomainQualifiedName, kerberos, entity, persistent, transient

Default value: transient

NameIDExpr

Expression that will be evaluated to obtain NameIdentifier to be sent in assertion

Attribute1

Name of attribute1 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute1Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute1FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute1Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute2

Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute2Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute2FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute2Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute3

Name of attribute3 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute3Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute3FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute3Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute4

Name of attribute4 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute4Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute4FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute4Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute5

Name of attribute5 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute5Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute5FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute5Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute6

Name of attribute6 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute6Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute6FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute6Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute7

Name of attribute7 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute7Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute7FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute7Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute8

Name of attribute8 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute8Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute8FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute8Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute9

Name of attribute9 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute9Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute9FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute9Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute10

Name of attribute10 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute10Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute10FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute10Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute11

Name of attribute11 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute11Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute11FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute11Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute12

Name of attribute12 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute12Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute12FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute12Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute13

Name of attribute13 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute13Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute13FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute13Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute14

Name of attribute14 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute14Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute14FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute14Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute15

Name of attribute15 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute15Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute15FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute15Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute16

Name of attribute16 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute16Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute16FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute16Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

rm tm samlSSOProfile

Deletes an existing saml single sign-on traffic profile.

Synopsys

```
rm tm samlSSOProfile <name>
```

Arguments

name

Name for the new saml single sign-on profile. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after an SSO action is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my action" or 'my action').

set tm samlSSOProfile

Modifies the specified attributes of a saml single sign-on traffic profile.

Synopsys

```
set tm samlSSOProfile <name> [-samlSigningCertName <string>] [-assertionConsumerServiceURL <URL>] [-  
sendPassword ( ON | OFF )] [-samlIssuerName <string>] [-relaystateRule <expression>] [-signatureAlg ( RSA-SHA1  
| RSA-SHA256 )] [-digestMethod ( SHA1 | SHA256 )] [-audience <string>] [-NameIDFormat <NameIDFormat>] [-  
NameIDExpr <string>] [-Attribute1 <string> -Attribute1Expr <string> [-Attribute1FriendlyName <string>] [-  
Attribute1Format ( URI | Basic )]] [-Attribute2 <string> -Attribute2Expr <string> [-Attribute2FriendlyName <string>] [-  
Attribute2Format ( URI | Basic )]] [-Attribute3 <string> -Attribute3Expr <string> [-Attribute3FriendlyName <string>] [-  
Attribute3Format ( URI | Basic )]] [-Attribute4 <string> -Attribute4Expr <string> [-Attribute4FriendlyName <string>] [-  
Attribute4Format ( URI | Basic )]] [-Attribute5 <string> -Attribute5Expr <string> [-Attribute5FriendlyName <string>] [-  
Attribute5Format ( URI | Basic )]] [-Attribute6 <string> -Attribute6Expr <string> [-Attribute6FriendlyName <string>] [-  
Attribute6Format ( URI | Basic )]] [-Attribute7 <string> -Attribute7Expr <string> [-Attribute7FriendlyName <string>] [-  
Attribute7Format ( URI | Basic )]] [-Attribute8 <string> -Attribute8Expr <string> [-Attribute8FriendlyName <string>] [-  
Attribute8Format ( URI | Basic )]] [-Attribute9 <string> -Attribute9Expr <string> [-Attribute9FriendlyName <string>] [-  
Attribute9Format ( URI | Basic )]] [-Attribute10 <string> -Attribute10Expr <string> [-Attribute10FriendlyName  
<string>] [-Attribute10Format ( URI | Basic )]] [-Attribute11 <string> -Attribute11Expr <string> [-  
Attribute11FriendlyName <string>] [-Attribute11Format ( URI | Basic )]] [-Attribute12 <string> -Attribute12Expr  
<string> [-Attribute12FriendlyName <string>] [-Attribute12Format ( URI | Basic )]] [-Attribute13 <string> -  
Attribute13Expr <string> [-Attribute13FriendlyName <string>] [-Attribute13Format ( URI | Basic )]] [-Attribute14  
<string> -Attribute14Expr <string> [-Attribute14FriendlyName <string>] [-Attribute14Format ( URI | Basic )]] [-  
Attribute15 <string> -Attribute15Expr <string> [-Attribute15FriendlyName <string>] [-Attribute15Format ( URI | Basic  
)]] [-Attribute16 <string> -Attribute16Expr <string> [-Attribute16FriendlyName <string>] [-Attribute16Format ( URI |  
Basic )]]
```

Arguments

name

Name for the new saml single sign-on profile. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after an SSO action is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my action" or 'my action').

samlSigningCertName

Name of the signing authority as given in the SAML server's SSL certificate.

assertionConsumerServiceURL

URL to which the assertion is to be sent.

sendPassword

Option to send password in assertion.

Possible values: ON, OFF

Default value: OFF

samlIssuerName

The name to be used in requests sent from Netscaler to IdP to uniquely identify Netscaler.

relaystateRule

Expression to extract relaystate to be sent along with assertion. Evaluation of this expression should return TEXT content. This is typically a targ

et url to which user is redirected after the recipient validates SAML token

signatureAlg

Algorithm to be used to sign/verify SAML transactions

Possible values: RSA-SHA1, RSA-SHA256

Default value: RSA-SHA1

digestMethod

Algorithm to be used to compute/verify digest for SAML transactions

Possible values: SHA1, SHA256

Default value: SHA1

audience

Audience for which assertion sent by IdP is applicable. This is typically entity name or url that represents ServiceProvider

Maximum value: 256

NameIDFormat

Format of Name Identifier sent in Assertion.

Possible values: Unspecified, emailAddress, X509SubjectName, WindowsDomainQualifiedName, kerberos, entity, persistent, transient

Default value: transient

NameIDExpr

Expression that will be evaluated to obtain NameIdentifier to be sent in assertion

Attribute1

Name of attribute1 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute1Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute1FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute1Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute2

Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute2Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute2FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute2Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute3

Name of attribute3 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute3Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute3FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute3Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute4

Name of attribute4 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute4Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute4FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute4Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute5

Name of attribute5 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute5Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute5FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute5Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute6

Name of attribute6 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute6Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute6FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute6Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute7

Name of attribute7 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute7Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute7FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute7Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute8

Name of attribute8 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute8Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute8FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute8Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute9

Name of attribute9 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute9Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute9FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute9Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute10

Name of attribute10 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute10Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute10FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute10Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute11

Name of attribute11 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute11Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute11FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute11Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute12

Name of attribute12 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute12Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute12FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute12Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute13

Name of attribute13 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute13Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute13FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute13Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute14

Name of attribute14 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute14Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute14FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute14Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute15

Name of attribute15 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute15Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute15FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute15Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute16

Name of attribute16 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute16Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute16FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute16Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

unset tm samlSSOProfile

Use this command to remove tm samlSSOProfile settings. Refer to the set tm samlSSOProfile command for meanings of the arguments.

Synopsis

```
unset tm samlSSOProfile <name> [-samlSigningCertName] [-sendPassword] [-samlIssuerName] [-signatureAlg] [-digestMethod] [-audience] [-NameIDFormat] [-NameIDExpr] [-Attribute1FriendlyName] [-Attribute1Format] [-Attribute2FriendlyName] [-Attribute2Format] [-Attribute3FriendlyName] [-Attribute3Format] [-Attribute4FriendlyName] [-Attribute4Format] [-Attribute5FriendlyName] [-Attribute5Format] [-Attribute6FriendlyName] [-Attribute6Format] [-Attribute7FriendlyName] [-Attribute7Format] [-Attribute8FriendlyName] [-Attribute8Format] [-Attribute9FriendlyName] [-Attribute9Format] [-Attribute10FriendlyName] [-Attribute10Format] [-Attribute11FriendlyName] [-Attribute11Format] [-Attribute12FriendlyName] [-Attribute12Format] [-Attribute13FriendlyName] [-Attribute13Format] [-Attribute14FriendlyName] [-Attribute14Format] [-Attribute15FriendlyName] [-Attribute15Format] [-Attribute16FriendlyName] [-Attribute16Format]
```

show tm samlSSOProfile

Displays information about all configured saml single sign-on profiles, or displays detailed information about the specified action.

Synopsis

```
show tm samlSSOProfile [<name>]
```

Arguments

name

Name for the new saml single sign-on profile. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after an SSO action is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my action" or 'my action').

Outputs

samlSigningCertName

Name of the signing authority as given in the SAML server's SSL certificate.

assertionConsumerServiceURL

URL to which the assertion is to be sent.

sendPassword

Option to send password in assertion.

samlIssuerName

The name to be used in requests sent from Netscaler to IdP to uniquely identify Netscaler.

relaystateRule

Expression to extract relaystate to be sent along with assertion. Evaluation of this expression should return TEXT content. This is typically a targ

et url to which user is redirected after the recipient validates SAML token

signatureAlg

Algorithm to be used to sign/verify SAML transactions

digestMethod

Algorithm to be used to compute/verify digest for SAML transactions

audience

Audience for which assertion sent by IdP is applicable. This is typically entity name or url that represents ServiceProvider

NameIDFormat

Format of Name Identifier sent in Assertion.

NameIDExpr

Expression that will be evaluated to obtain NameIdentifier to be sent in assertion

Attribute1

Name of attribute1 that needs to be sent in SAML Assertion

Attribute2

Name of attribute2 that needs to be sent in SAML Assertion

Attribute3

Name of attribute3 that needs to be sent in SAML Assertion

Attribute4

Name of attribute4 that needs to be sent in SAML Assertion

Attribute5

Name of attribute5 that needs to be sent in SAML Assertion

Attribute6

Name of attribute6 that needs to be sent in SAML Assertion

Attribute7

Name of attribute7 that needs to be sent in SAML Assertion

Attribute8

Name of attribute8 that needs to be sent in SAML Assertion

Attribute9

Name of attribute9 that needs to be sent in SAML Assertion

Attribute10

Name of attribute10 that needs to be sent in SAML Assertion

Attribute11

Name of attribute11 that needs to be sent in SAML Assertion

Attribute12

Name of attribute12 that needs to be sent in SAML Assertion

Attribute13

Name of attribute13 that needs to be sent in SAML Assertion

Attribute14

Name of attribute14 that needs to be sent in SAML Assertion

Attribute15

Name of attribute15 that needs to be sent in SAML Assertion

Attribute16

Name of attribute16 that needs to be sent in SAML Assertion

Attribute1FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute2FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute3FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute4FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute5FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute6FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute7FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute8FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute9FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute10FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute11FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute12FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute13FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute14FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute15FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute16FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute1Format

Format of Attribute1 to be sent in Assertion.

Attribute2Format

Format of Attribute1 to be sent in Assertion.

Attribute3Format

Format of Attribute1 to be sent in Assertion.

Attribute4Format

Format of Attribute1 to be sent in Assertion.

Attribute5Format

Format of Attribute1 to be sent in Assertion.

Attribute6Format

Format of Attribute1 to be sent in Assertion.

Attribute7Format

Format of Attribute1 to be sent in Assertion.

Attribute8Format

Format of Attribute1 to be sent in Assertion.

Attribute9Format

Format of Attribute1 to be sent in Assertion.

Attribute10Format

Format of Attribute1 to be sent in Assertion.

Attribute11Format

Format of Attribute1 to be sent in Assertion.

Attribute12Format

Format of Attribute1 to be sent in Assertion.

Attribute13Format

Format of Attribute1 to be sent in Assertion.

Attribute14Format

Format of Attribute1 to be sent in Assertion.

Attribute15Format

Format of Attribute1 to be sent in Assertion.

Attribute16Format

Format of Attribute1 to be sent in Assertion.

Attribute1Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute2Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute3Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute4Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute5Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute6Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute7Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute8Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute9Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute10Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute11Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute12Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute13Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute14Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute15Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute16Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

devno

count

stateflag

tm sessionAction

The following operations can be performed on "tm sessionAction":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add tm sessionAction

Creates a session action (profile) that allows you to override global settings for any of the session parameters.

Synopsys

```
add tm sessionAction <name> [-sessTimeout <mins>] [-defaultAuthorizationAction ( ALLOW | DENY )] [-SSO ( ON | OFF )] [-ssoCredential ( PRIMARY | SECONDARY )] [-ssoDomain <string>] [-httpOnlyCookie ( YES | NO )] [-kcdAccount <string>] [-persistentCookie ( ON | OFF )] [-persistentCookieValidity <mins>] [-homePage <URL>]
```

Arguments

name

Name for the session action. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after a session action is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my action" or 'my action').

sessTimeout

Session timeout, in minutes. If there is no traffic during the timeout period, the user is disconnected and must reauthenticate to access intranet resources.

Minimum value: 1

defaultAuthorizationAction

Allow or deny access to content for which there is no specific authorization policy.

Possible values: ALLOW, DENY

SSO

Use single sign-on (SSO) to log users on to all web applications automatically after they authenticate, or pass users to the web application logon page to authenticate to each application individually.

Possible values: ON, OFF

Default value: OFF

ssoCredential

Use the primary or secondary authentication credentials for single sign-on (SSO).

Possible values: PRIMARY, SECONDARY

ssoDomain

Domain to use for single sign-on (SSO).

httpOnlyCookie

Allow only an HTTP session cookie, in which case the cookie cannot be accessed by scripts.

Possible values: YES, NO

kcdAccount

Kerberos constrained delegation account name

persistentCookie

Enable or disable persistent SSO cookies for the traffic management (TM) session. A persistent cookie remains on the user device and is sent with each HTTP request. The cookie becomes stale if the session ends. This setting is overwritten if a traffic action sets persistent cookie to OFF.

Note: If persistent cookie is enabled, make sure you set the persistent cookie validity.

Possible values: ON, OFF

persistentCookieValidity

Integer specifying the number of minutes for which the persistent cookie remains valid. Can be set only if the persistent cookie setting is enabled.

Minimum value: 1

homePage

Web address of the home page that a user is displayed when authentication vserver is bookmarked and used to login.

rm tm sessionAction

Deletes an existing session action.

Synopsis

```
rm tm sessionAction <name>
```

Arguments

name

Name of the session action to delete.

set tm sessionAction

Modifies the specified parameters of an existing session action.

Synopsis

```
set tm sessionAction <name> [-sessTimeout <mins>] [-defaultAuthorizationAction ( ALLOW | DENY )] [-SSO ( ON | OFF )] [-ssoCredential ( PRIMARY | SECONDARY )] [-ssoDomain <string>] [-kcdAccount <string>] [-httpOnlyCookie ( YES | NO )] [-persistentCookie ( ON | OFF )] [-persistentCookieValidity <positive_integer>] [-homePage <URL>]
```

Arguments

name

Name of the session action to modify.

sessTimeout

Session timeout, in minutes. If there is no traffic during the timeout period, the user is disconnected and must reauthenticate to access intranet resources.

Minimum value: 1

defaultAuthorizationAction

Allow or deny access to content for which there is no specific authorization policy.

Possible values: ALLOW, DENY

SSO

Use single sign-on (SSO) to log users on to all web applications automatically after they authenticate, or pass users to the web application logon page to authenticate to each application individually.

Possible values: ON, OFF

Default value: OFF

ssoCredential

Use the primary or secondary authentication credentials for single sign-on (SSO).

Possible values: PRIMARY, SECONDARY

ssoDomain

Domain to use for single sign-on (SSO).

kcdAccount

Kerberos constrained delegation account name

httpOnlyCookie

Allow only an HTTP session cookie, in which case the cookie cannot be accessed by scripts.

Possible values: YES, NO

persistentCookie

Enable or disable persistent SSO cookies for the traffic management (TM) session. A persistent cookie remains on the user device and is sent with each HTTP request. The cookie becomes stale if the session ends. This setting is overwritten if a traffic action sets persistent cookie to OFF.

Note: If persistent cookie is enabled, make sure you set the persistent cookie validity.

Possible values: ON, OFF

persistentCookieValidity

Integer specifying the number of minutes for which the persistent cookie remains valid. Can be set only if the persistent cookie setting is enabled.

Minimum value: 1

homePage

Web address of the home page that a user is displayed when authentication vserver is bookmarked and used to login.

unset tm sessionAction

Use this command to remove tm sessionAction settings. Refer to the set tm sessionAction command for meanings of the arguments.

Synopsys

```
unset tm sessionAction <name> [-sessTimeout] [-defaultAuthorizationAction] [-SSO] [-ssoCredential] [-ssoDomain] [-kcdAccount] [-httpOnlyCookie] [-persistentCookie] [-persistentCookieValidity] [-homePage]
```

show tm sessionAction

Displays information about all configured traffic management (TM) session actions, or detailed information about the specified TM session action.

Synopsys

show tm sessionAction [<name>]

Arguments

name

Name of the existing traffic management (TM) session action for which to display detailed information.

Outputs

sessTimeout

The session timeout, in minutes, set by the action.

defaultAuthorizationAction

The Authorization Action, e.g. allow or deny

stateflag

SSO

Whether or not Single Sign-On is used for this session.

ssoCredential

Use the primary or secondary authentication credentials for single sign-on (SSO).

ssoDomain

Domain to use for single sign-on (SSO).

kcdAccount

Kerberos constrained delegation account name

httpOnlyCookie

Allow only an HTTP session cookie, in which case the cookie cannot be accessed by scripts.

persistentCookie

Enable or disable persistent SSO cookies for the traffic management (TM) session. A persistent cookie remains on the user device and is sent with each HTTP request. The cookie becomes stale if the session ends. This setting is overwritten if a traffic action sets persistent cookie to OFF.

Note: If persistent cookie is enabled, make sure you set the persistent cookie validity.

persistentCookieValidity

Integer specifying the number of minutes for which the persistent cookie remains valid. Can be set only if the persistent cookie setting is enabled.

homePage

Web address of the home page that a user is displayed when authentication vserver is bookmarked and used to login.

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

devno

count

tm sessionParameter

The following operations can be performed on "tm sessionParameter":

[set](#) | [unset](#) | [show](#)

set tm sessionParameter

Sets global parameters for the traffic management (TM) session. Parameters defined when adding a traffic session action override these parameters.

Synopsys

```
set tm sessionParameter [-sessTimeout <mins>] [-defaultAuthorizationAction ( ALLOW | DENY )] [-SSO ( ON | OFF )] [-ssoCredential ( PRIMARY | SECONDARY )] [-ssoDomain <string>] [-kcdAccount <string>] [-httpOnlyCookie ( YES | NO )] [-persistentCookie ( ON | OFF )] [-persistentCookieValidity <positive_integer>] [-homePage <URL>]
```

Arguments

sessTimeout

Session timeout, in minutes. If there is no traffic during the timeout period, the user is disconnected and must reauthenticate to access the intranet resources.

Default value: 30

Minimum value: 1

defaultAuthorizationAction

Allow or deny access to content for which there is no specific authorization policy.

Possible values: ALLOW, DENY

Default value: ALLOW

SSO

Log users on to all web applications automatically after they authenticate, or pass users to the web application logon page to authenticate for each application.

Possible values: ON, OFF

Default value: OFF

ssoCredential

Use primary or secondary authentication credentials for single sign-on.

Possible values: PRIMARY, SECONDARY

Default value: PRIMARY

ssoDomain

Domain to use for single sign-on.

kcdAccount

Kerberos constrained delegation account name

httpOnlyCookie

Allow only an HTTP session cookie, in which case the cookie cannot be accessed by scripts.

Possible values: YES, NO

Default value: YES

persistentCookie

Use persistent SSO cookies for the traffic session. A persistent cookie remains on the user device and is sent with each HTTP request. The cookie becomes stale if the session ends.

Possible values: ON, OFF

Default value: OFF

persistentCookieValidity

Integer specifying the number of minutes for which the persistent cookie remains valid. Can be set only if the persistence cookie setting is enabled.

Minimum value: 1

homePage

Web address of the home page that a user is displayed when authentication vserver is bookmarked and used to login.

Default value: "None"

unset tm sessionParameter

Resets the attributes of the specified traffic session parameters. Attributes for which a default value is available revert to their default values. Refer to the set tm sessionParameter command for descriptions of the parameters..Refer to the set tm sessionParameter command for meanings of the arguments.

Synopsis

unset tm sessionParameter [-sessTimeout] [-SSO] [-ssoDomain] [-kcdAccount] [-persistentCookie] [-homePage] [-defaultAuthorizationAction] [-ssoCredential] [-httpOnlyCookie] [-persistentCookieValidity]

show tm sessionParameter

Displays information about traffic session parameters.

Synopsis

show tm sessionParameter

Outputs

name

sessTimeout

The session timeout, in minutes.

defaultAuthorizationAction

The Authentication Action, e.g. allow or deny.

SSO

Whether or not Single Sign-On is used for this session.

ssoCredential

Use primary or secondary authentication credentials for single sign-on.

ssoDomain

Domain to use for single sign-on.

kcdAccount

Kerberos constrained delegation account name

httpOnlyCookie

Allow only an HTTP session cookie, in which case the cookie cannot be accessed by scripts.

homePage

Web address of the home page that a user is displayed when authentication vserver is bookmarked and used to login.

persistentCookie

Use persistent SSO cookies for the traffic session. A persistent cookie remains on the user device and is sent with each HTTP request. The cookie becomes stale if the session ends.

persistentCookieValidity

Integer specifying the number of minutes for which the persistent cookie remains valid. Can be set only if the persistence cookie setting is enabled.

tm sessionPolicy

The following operations can be performed on "tm sessionPolicy":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add tm sessionPolicy

Creates a traffic management (TM) session policy, which is applied after the user logs on to the AAA virtual server, to customize user sessions.

Synopsys

add tm sessionPolicy <name> <rule> <action>

Arguments

name

Name for the session policy. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at sign (@), equal sign (=), and hyphen (-) characters. Cannot be changed after a session policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my policy" or 'my policy').

rule

Expression, against which traffic is evaluated. Written in the classic syntax.

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

The following requirements apply only to the NetScaler CLI:

- * If the expression includes one or more spaces, enclose the entire expression in double quotation marks.
- * If the expression itself includes double quotation marks, escape the quotations by using the \\ character.
- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Action to be applied to connections that match this policy.

rm tm sessionPolicy

Removes an existing traffic management (TM) session policy.

Synopsys

rm tm sessionPolicy <name>

Arguments

name

Name of the session policy to remove.

set tm sessionPolicy

Modifies the rule or action of an existing traffic management (TM) session policy.

Synopsys

```
set tm sessionPolicy <name> [-rule <expression>] [-action <string>]
```

Arguments

name

Name of the session policy to modify.

rule

Expression, against which traffic is evaluated. Written in the classic syntax.

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

The following requirements apply only to the NetScaler CLI:

- * If the expression includes one or more spaces, enclose the entire expression in double quotation marks.
- * If the expression itself includes double quotation marks, escape the quotations by using the \ character.
- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Action to be applied to connections that match this policy.

unset tm sessionPolicy

Use this command to remove tm sessionPolicy settings. Refer to the set tm sessionPolicy command for meanings of the arguments.

Synopsys

```
unset tm sessionPolicy <name> [-rule] [-action]
```

show tm sessionPolicy

Displays information about all the configured traffic management (TM) session policies, or displays detailed information about the specified TM session policy.

Synopsys

```
show tm sessionPolicy [<name>]
```

Arguments

name

Name of the session policy for which to display detailed information.

Outputs

rule

Expression, against which traffic is evaluated. Written in the classic syntax.

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

The following requirements apply only to the NetScaler CLI:

- * If the expression includes one or more spaces, enclose the entire expression in double quotation marks.
- * If the expression itself includes double quotation marks, escape the quotations by using the \\ character.
- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Action to be applied to connections that match this policy.

boundTo

The entity name to which policy is bound

activePolicy

priority

bindPolicyType

policyType

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

devno

count

stateflag

tm trafficAction

The following operations can be performed on "tm trafficAction":

add | **rm** | **set** | **unset** | **show**

add tm trafficAction

Creates a traffic action to set traffic characteristics at run time. You can create a traffic action for an application that is installed in the internal network (for example, an action that defines the destination IP address and destination port, and sets the amount of time a user can stay logged on to the application, such as 15 minutes).

Synopsys

```
add tm trafficAction <name> [-appTimeout <mins>] [-SSO ( ON | OFF )] [-formSSOAction <string>]] [-persistentCookie ( ON | OFF )] [-InitiateLogout ( ON | OFF )] [-kcdAccount <string>] [-samlSSOProfile <string>] [-forcedTimeout <forcedTimeout> -forcedTimeoutVal <mins> ] [-userExpression <string>] [-passwdExpression <string>]
```

Arguments

name

Name for the traffic action. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after a traffic action is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my action" or 'my action').

appTimeout

Time interval, in minutes, of user inactivity after which the connection is closed.

Minimum value: 1

Maximum value: 715827

SSO

Use single sign-on for the resource that the user is accessing now.

Possible values: ON, OFF

formSSOAction

Name of the configured form-based single sign-on profile.

persistentCookie

Use persistent cookies for the traffic session. A persistent cookie remains on the user device and is sent with each HTTP request. The cookie becomes stale if the session ends.

Possible values: ON, OFF

InitiateLogout

Initiate logout for the traffic management (TM) session if the policy evaluates to true. The session is then terminated after two minutes.

Possible values: ON, OFF

kcdAccount

Kerberos constrained delegation account name

Default value: "None"

samlSSOProfile

Profile to be used for doing SAML SSO to remote relying party

forcedTimeout

Setting to start, stop or reset TM session force timer

Possible values: START, STOP, RESET

forcedTimeoutVal

Time interval, in minutes, for which force timer should be set.

userExpression

expression that will be evaluated to obtain username for SingleSignOn

Maximum value: 256

passwdExpression

expression that will be evaluated to obtain password for SingleSignOn

Maximum value: 256

rm tm trafficAction

Removes an existing traffic action.

Synopsis

```
rm tm trafficAction <name>
```

Arguments

name

Name of the traffic action to remove.

set tm trafficAction

Modifies the specified parameters of an existing traffic action.

Synopsis

```
set tm trafficAction <name> [-appTimeout <mins>] [-SSO ( ON | OFF )] [-formSSOAction <string>] [-persistentCookie ( ON | OFF )] [-InitiateLogout ( ON | OFF )] [-kcdAccount <string>] [-samlSSOProfile <string>] [-forcedTimeout <forcedTimeout>] [-forcedTimeoutVal <mins>] [-userExpression <string>] [-passwdExpression <string>]
```

Arguments

name

Name of the traffic action to modify.

appTimeout

Time interval, in minutes, of user inactivity after which the connection is closed.

Minimum value: 1

Maximum value: 715827

SSO

Use single sign-on for the resource that the user is accessing now.

Possible values: ON, OFF

formSSOAction

Name of the configured form-based single sign-on profile.

persistentCookie

Use persistent cookies for the traffic session. A persistent cookie remains on the user device and is sent with each HTTP request. The cookie becomes stale if the session ends.

Possible values: ON, OFF

InitiateLogout

Initiate logout for the traffic management (TM) session if the policy evaluates to true. The session is then terminated after two minutes.

Possible values: ON, OFF

kcdAccount

Kerberos constrained delegation account name

Default value: "None"

samlSSOProfile

Profile to be used for doing SAML SSO to remote relying party

forcedTimeout

Setting to start, stop or reset TM session force timer

Possible values: START, STOP, RESET

forcedTimeoutVal

Time interval, in minutes, for which force timer should be set.

userExpression

expression that will be evaluated to obtain username for SingleSignOn

Maximum value: 256

passwdExpression

expression that will be evaluated to obtain password for SingleSignOn

Maximum value: 256

unset tm trafficAction

Use this command to remove tm trafficAction settings. Refer to the set tm trafficAction command for meanings of the arguments.

Synopsys

```
unset tm trafficAction <name> [-persistentCookie] [-kcdAccount] [-forcedTimeout] [-userExpression] [-passwdExpression]
```

show tm trafficAction

Displays information about all configured traffic management (TM) traffic actions, or displays detailed information about the specified TM traffic action.

Synopsys

show tm trafficAction [<name>]

Arguments

name

Name of the traffic action for which to display detailed information.

Outputs

appTimeout

The application timeout

SSO

Whether or not Single Sign On is enabled.

formSSOAction

Name of the configured form-based single sign-on profile.

stateflag

persistentCookie

Use persistent cookies for the traffic session. A persistent cookie remains on the user device and is sent with each HTTP request. The cookie becomes stale if the session ends.

InitiateLogout

Whether Logout is initiated with this action

kcdAccount

Kerberos constrained delegation account name

samlSSOProfile

Profile to be used for doing SAML SSO to remote relying party

forcedTimeout

Setting to start, stop or reset TM session force timer

forcedTimeoutVal

Time interval, in minutes, for which force timer should be set.

userExpression

expression that will be evaluated to obtain username for SingleSignOn

passwdExpression

expression that will be evaluated to obtain password for SingleSignOn

devno

count

tm trafficPolicy

The following operations can be performed on "tm trafficPolicy":

add | **rm** | **set** | **unset** | **show** | **stat**

add tm trafficPolicy

Adds a traffic policy to use for setting connection timeout, single sign-on, and initiating logout. The policy sets the characteristics of application traffic at run time.

Synopsis

add tm trafficPolicy <name> <rule> <action>

Arguments

name

Name for the traffic policy. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my policy" or 'my policy').

rule

Expression, against which traffic is evaluated. Written in the classic syntax.

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

The following requirements apply only to the NetScaler CLI:

- * If the expression includes one or more spaces, enclose the entire expression in double quotation marks.
- * If the expression itself includes double quotation marks, escape the quotations by using the \\ character.
- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Name of the action to apply to requests or connections that match this policy.

rm tm trafficPolicy

Removes an existing traffic policy.

Synopsis

rm tm trafficPolicy <name>

Arguments

name

Name of the traffic policy to remove.

set tm trafficPolicy

Modifies the specified parameters of an existing traffic policy.

Synopsis

```
set tm trafficPolicy <name> [-rule <expression>] [-action <string>]
```

Arguments

name

Name of the traffic policy to modify.

rule

Expression, against which traffic is evaluated. Written in the classic syntax.

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

The following requirements apply only to the NetScaler CLI:

- * If the expression includes one or more spaces, enclose the entire expression in double quotation marks.
- * If the expression itself includes double quotation marks, escape the quotations by using the \ character.
- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Name of the action to apply to requests or connections that match this policy.

unset tm trafficPolicy

Use this command to remove tm trafficPolicy settings. Refer to the set tm trafficPolicy command for meanings of the arguments.

Synopsis

```
unset tm trafficPolicy <name> [-rule] [-action]
```

show tm trafficPolicy

Displays information about all configured traffic management (TM) traffic policies, or displays detailed information about the specified TM traffic policy.

Synopsis

```
show tm trafficPolicy [<name>]
```

Arguments

name

Name of the traffic policy for which to display detailed information.

Outputs

rule

The rule used by the vpn traffic policy. Rules are combinations of Expressions. Expressions are simple conditions, such as a test for equality, applied to operands, such as a URL string or an IP address. Expression syntax is described in the Installation and Configuration Guide

action

The action to be performed when the rule is matched.

boundTo

The entity name to which policy is bound

activePolicy

priority

hits

Number of hits.

bindPolicyType

vserverType

stateflag

devno

count

stat tm trafficPolicy

Display Traffic Management traffic policy statistics.

Synopsys

```
stat tm trafficPolicy [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

name

The name of the TM traffic policy for which statistics will be displayed. If not given statistics are shown for all policies.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Policy hits (Hits)

Number of hits on the policy

Example

```
stat tm trafficpolicy.
```

Transform Commands

The entities on which you can perform NetScaler CLI operations:

- transform action
- transform global
- transform policy
- transform policylabel
- transform profile

transform action

The following operations can be performed on "transform action":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add transform action

Creates a URL Transformation action, which defines how a specific element in URLs in the request or response is to be modified. NOTE: In the URL Transformation feature (unlike all other NetScaler features), ?profile? and ?action? are not synonymous but refer to distinct entities. You must create the profile first, and then the actions.

Synopsys

```
add transform action <name> <profileName> <priority> [-state ( ENABLED | DISABLED )]
```

Arguments

name

Name for the URL transformation action.

Must begin with a letter, number, or the underscore character (`_`), and must contain only letters, numbers, and the hyphen (`-`), period (`.`), pound (`#`), space (), at (`@`), equals (`=`), colon (`:`), and underscore characters. Cannot be changed after the URL Transformation action is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my transform action? or ?my transform action).

profileName

Name of the URL Transformation profile with which to associate this action.

priority

Positive integer specifying the priority of the action within the profile. A lower number specifies a higher priority. Must be unique within the list of actions bound to the profile. Policies are evaluated in the order of their priority numbers, and the first policy that matches is applied.

Minimum value: 1

Maximum value: 2147483647

state

Enable or disable this action.

Possible values: ENABLED, DISABLED

Default value: ENABLED

rm transform action

Removes a URL Transformation action.

Synopsys

```
rm transform action <name>
```

Arguments

name

Name of the action.

set transform action

Modifies the settings of the specified URL Transformation action.

Synopsis

```
set transform action <name> [-priority <positive_integer>] [-reqUrlFrom <expression>] [-reqUrlInto <expression>] [-resUrlFrom <expression>] [-resUrlInto <expression>] [-cookieDomainFrom <expression>] [-cookieDomainInto <expression>] [-state ( ENABLED | DISABLED )] [-comment <string>]
```

Arguments

name

Name of the URL Transformation action to modify.

priority

Positive integer specifying the priority of the action within the profile. A lower number specifies a higher priority. Must be unique within the list of actions bound to the profile. Policies are evaluated in the order of their priority numbers, and the first policy that matches is applied.

Minimum value: 1

Maximum value: 2147483647

reqUrlFrom

PCRE-format regular expression that describes the request URL pattern to be transformed.

reqUrlInto

PCRE-format regular expression that describes the transformation to be performed on URLs that match the reqUrlFrom pattern.

resUrlFrom

PCRE-format regular expression that describes the response URL pattern to be transformed.

resUrlInto

PCRE-format regular expression that describes the transformation to be performed on URLs that match the resUrlFrom pattern.

cookieDomainFrom

Pattern that matches the domain to be transformed in Set-Cookie headers.

cookieDomainInto

PCRE-format regular expression that describes the transformation to be performed on cookie domains that match the cookieDomainFrom pattern.

NOTE: The cookie domain to be transformed is extracted from the request.

state

Enable or disable this action.

Possible values: ENABLED, DISABLED

Default value: ENABLED

comment

Any comments to preserve information about this URL Transformation action.

unset transform action

Use this command to remove transform action settings. Refer to the set transform action command for meanings of the arguments.

Synopsys

```
unset transform action <name> [-reqUrlFrom] [-reqUrlInto] [-resUrlFrom] [-resUrlInto] [-cookieDomainFrom] [-cookieDomainInto] [-state] [-comment]
```

show transform action

Displays a list of all URL Transformation actions currently assigned to the specified profile.

Synopsys

```
show transform action [<name>]
```

Arguments

name

Name of the profile.

Outputs

stateflag

profileName

Name of the URL Transformation profile with which to associate this action.

priority

Positive integer specifying the priority of the action within the profile. A lower number specifies a higher priority. Must be unique within the list of actions bound to the profile. Policies are evaluated in the order of their priority numbers, and the first policy that matches is applied.

reqUrlFrom

PCRE-format regular expression that describes the request URL pattern to be transformed.

reqUrlInto

PCRE-format regular expression that describes the transformation to be performed on URLs that match the reqUrlFrom pattern.

resUrlFrom

PCRE-format regular expression that describes the response URL pattern to be transformed.

resUrlInto

PCRE-format regular expression that describes the transformation to be performed on URLs that match the resUrlFrom pattern.

cookieDomainFrom

Pattern that matches the domain to be transformed in Set-Cookie headers.

cookieDomainInto

PCRE-format regular expression that describes the transformation to be performed on cookie domains that match the cookieDomainFrom pattern.

NOTE: The cookie domain to be transformed is extracted from the request.

continueMatching

Continue transforming using the next rule in the list.

state

Enable or disable this action.

comment

Any comments to preserve information about this URL Transformation action.

devno

count

transform global

The following operations can be performed on "transform global":

[bind](#) | [unbind](#) | [show](#)

bind transform global

Activates the specified URL Transformation policy for all traffic received by this NetScaler appliance. If you set policyName to a name that does not match an existing URL Transformation policy name, this command creates the policy, with the configuration that you specify.

Synopsys

```
bind transform global <policyName> <priority> [<gotoPriorityExpression>] [-type ( REQ_OVERRIDE | REQ_DEFAULT )] [-invoke (<labelType> <labelName>)]
```

Arguments

policyName

Name of the policy.

If you want to create the policy as well as activate it, specify a name for the policy. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.), pound (#), space (), at (@), equals (=), colon (:), and underscore characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my transform policy" or 'my transform policy').

priority

Positive integer specifying the priority of the policy within the list of globally-bound URL transformation policies. A lower number specifies a higher priority. Must be unique within the list. Policies are evaluated in the order of their priorities, and the first policy that matches is applied.

Minimum value: 1

Maximum value: 2147483647

gotoPriorityExpression

Optional expression or other value specifying the next policy to evaluate if the current policy evaluates to TRUE. Specify one of the following values:

- * NEXT ? Evaluate the policy with the next higher priority number.
- * END ? End policy evaluation.
- * USE_INVOCATION_RESULT ? Applicable if this policy invokes another policy label. If the final goto in the invoked policy label has a value of END, the evaluation stops. If the final goto is anything other than END, the current policy label performs a NEXT.
- * A PCRE-compatible regular expression that evaluates to a number.

If you specify an expression, the number to which it evaluates determines the next policy to evaluate, as follows:

- * If the expression evaluates to a higher numbered priority, the policy with that priority is evaluated next.
- * If the expression evaluates to the priority of the current policy, the policy with the next higher priority number is evaluated next.
- * If the expression evaluates to a number that is larger than the largest priority number, policy evaluation ends.

An UNDEF event is triggered if:

- * The expression is invalid.
- * The expression evaluates to a priority number that is smaller than the current policy's priority number.
- * The expression evaluates to a priority number that is between the current policy's priority number (say, 30) and the highest priority number (say, 100), but does not match any configured priority number (for example, the expression evaluates to the number 85). This example assumes that the priority number increments by 10 for every successive policy, and therefore a priority number of 85 does not exist in the policy label.

type

Bind point to which to bind the policy. Available settings function as follows:

- * REQ_OVERRIDE. Request override. Binds the policy to the priority request queue.
- * REQ_DEFAULT. Binds the policy to the default request queue.

Possible values: REQ_OVERRIDE, REQ_DEFAULT

invoke

If the current policy evaluates to TRUE, terminate evaluation of policies bound to the current policy label, and then forwards the request or response to the specified virtual server or evaluates the specified policy label.

labelType

Type of invocation. Available settings function as follows:

- * reqvserver - Send the request to the specified request virtual server.
- * resvserver - Send the response to the specified response virtual server.
- * policylabel - Invoke the specified policy label.

Possible values: reqvserver, resvserver, policylabel

labelName

Name of the policy label to invoke if the current policy evaluates to TRUE, the invoke parameter is set, and the label type is Policy Label.

Example

```
bind transform global pol9 9
```

unbind transform global

Unbinds the specified URL Transformation policy from URL Transformation global.

Synopsys

```
unbind transform global <policyName> [-type ( REQ_OVERRIDE | REQ_DEFAULT )] [-priority <positive_integer>]
```

Arguments

policyName

The name of the policy to be unbound.

type

The bindpoint from which the policy is to be unbound.

Possible values: REQ_OVERRIDE, REQ_DEFAULT

priority

Priority of the NOPOLICY to be unbound.

Minimum value: 1

Maximum value: 2147483647

Example

```
unbind transform global pol9
```

show transform global

Displays the policies bound to the specified URL Transformation global bind point. If no bind point is specified, displays a list of all policies bound to URL Transformation global.

Synopsys

```
show transform global [-type ( REQ_OVERRIDE | REQ_DEFAULT )]
```

Arguments

type

Specifies the bind point to which to bind the policy. Available settings function as follows:

* REQ_OVERRIDE. Request override. Binds the policy to the priority request queue.

* REQ_DEFAULT. Binds the policy to the default request queue.

Possible values: REQ_OVERRIDE, REQ_DEFAULT

Outputs

stateflag

policyName

Name of the transform policy.

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

invoke

If the current policy evaluates to TRUE, terminate evaluation of policies bound to the current policy label, and then forwards the request or response to the specified virtual server or evaluates the specified policy label.

labelType

Type of invocation. Available settings function as follows:

* reqvserver - Send the request to the specified request virtual server.

* resvserver - Send the response to the specified response virtual server.

* policylabel - Invoke the specified policy label.

labelName

Name of the policy label to invoke if the current policy evaluates to TRUE, the invoke parameter is set, and the label type is Policy Label.

flowType

flowtype of the bound transform policy.

numpol

The number of policies bound to the bindpoint.

flags**devno****count**

Example

```
show transform global
```

transform policy

The following operations can be performed on "transform policy":

add | **rm** | **set** | **unset** | **show** | **stat** | **rename**

add transform policy

Creates a URL Transformation policy, which specifies the requests and responses to be transformed by the associated profile.

Synopsys

add transform policy <name> <rule> <profileName> [-comment <string>] [-logAction <string>]

Arguments

name

Name for the URL Transformation policy.

Must begin with a letter, number, or the underscore character (`_`), and must contain only letters, numbers, and the hyphen (`-`), period (`.`), pound (`#`), space (), at (`@`), equals (`=`), colon (`:`), and underscore characters. Can be changed after the URL Transformation policy is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, `?my transform policy?` or `?my transform policy`).

rule

Expression, or name of a named expression, against which to evaluate traffic. Can be written in either default or classic syntax. Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the `+` operator. For example, you can create a 500-character string as follows: `"<string of 255 characters>" + "<string of 245 characters>"`

The following requirements apply only to the NetScaler CLI:

- * If the expression includes blank spaces, the entire expression must be enclosed in double quotation marks.
- * If the expression itself includes double quotation marks, you must escape the quotations by using the `\` character.
- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

profileName

Name of the URL Transformation profile to use to transform requests and responses that match the policy.

comment

Any comments to preserve information about this URL Transformation policy.

logAction

Log server to use to log connections that match this policy.

rm transform policy

Removes the specified URL Transformation policy.

Synopsys

rm transform policy <name>

Arguments

name

Name of the policy to remove.

Example

```
rm transform policy trans_pol
```

set transform policy

Modifies the specified parameters of a URL Transformation policy.

Synopsys

```
set transform policy <name> [-rule <expression>] [-profileName <string>] [-comment <string>] [-logAction <string>]
```

Arguments

name

Name of the policy to modify.

rule

Expression, or name of a named expression, against which to evaluate traffic. Can be written in either default or classic syntax. Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

The following requirements apply only to the NetScaler CLI:

- * If the expression includes blank spaces, the entire expression must be enclosed in double quotation marks.
- * If the expression itself includes double quotation marks, you must escape the quotations by using the \ character.
- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

profileName

Name of the URL Transformation profile to use to transform requests and responses that match the policy.

comment

Any comments to preserve information about this URL Transformation policy.

logAction

Log server to use to log connections that match this policy.

Example

```
set transform policy pol9 -rule "HTTP.REQ.HEADER(\\\\"header\\\\" ).CONTAINS(\\\\"qh2\\\\" "
```

unset transform policy

Removes the settings of an existing URL Transformation policy. Attributes for which a default value is available revert to their default values. See the set transform policy command for a description of the parameters..Refer to the set transform policy command for meanings of the arguments.

Synopsys

`unset transform policy <name> [-comment] [-logAction]`

Example

```
unset transform policy pol9 -undefAction
```

show transform policy

Displays the current settings for the specified URL Transformation policy. If no policy name is specified, displays a list of all URL Transformation policies currently configured on the NetScaler appliance.

Synopsys

`show transform policy [<name>]`

Arguments

name

Name of the URL Transformation policy.

Outputs

stateflag

rule

Expression, or name of a named expression, against which to evaluate traffic. Can be written in either default or classic syntax. Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

The following requirements apply only to the NetScaler CLI:

- * If the expression includes blank spaces, the entire expression must be enclosed in double quotation marks.
- * If the expression itself includes double quotation marks, you must escape the quotations by using the \ character.
- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

profileName

Name of the URL Transformation profile to use to transform requests and responses that match the policy.

priority

Specifies the priority of the policy.

hits

Number of hits.

activePolicy

Indicates whether policy is bound or not.

boundTo

Location where policy is bound

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

labelType

Type of policy label invocation.

labelName

Name of the label to invoke if the current policy rule evaluates to TRUE.

comment

Any comments to preserve information about this URL Transformation policy.

logAction

Log server to use to log connections that match this policy.

bindPolicyType

isDefault

A value of true is returned if it is a default transform policy.

vserverType

devno

count

stat transform policy

Displays statistics for the specified URL Transformation policy. If no policy name is specified, displays abbreviated statistics for all URL Transformation policies currently configured on the NetScaler appliance.

Synopsys

```
stat transform policy [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

name

Name of the policy.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Policy hits (Hits)

Number of hits on the policy

Policy undef hits (Undefhits)

Number of undef hits on the policy

Example

```
stat transform policy
```

rename transform policy

Renames a URL Transformation policy.

Synopsys

```
rename transform policy <name>@ <newName>@
```

Arguments

name

Existing name of the policy.

newName

New name for the policy. Must begin with a letter, number, or the underscore character (`_`), and must contain only letters, numbers, and the hyphen (`-`), period (`.`) pound (`#`), space (), at (`@`), equals (`=`), colon (`:`), and underscore characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, `?my transform policy?` or `'my transform policy'`).

Example

```
rename transform policy oldname newname
```

transform policylabel

The following operations can be performed on "transform policylabel":

add | **rm** | **bind** | **unbind** | **show** | **stat** | **rename**

add transform policylabel

Creates a URL Transformation policy label. A policy label is a tool for evaluating a set of policies in a specified order. By using a policy label, you can configure the URL Transformation feature to choose the next policy, invoke a different policy label, or terminate policy evaluation completely by looking at whether the previous policy evaluated to TRUE or FALSE.

Synopsys

```
add transform policylabel <labelName> <policylabeltype>
```

Arguments

labelName

Name for the policy label. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.), pound (#), space (), at (@), equals (=), colon (:), and underscore characters. Can be changed after the URL Transformation policy label is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my transform policylabel" or 'my transform policylabel').

policylabeltype

Types of transformations allowed by the policies bound to the label. For URL transformation, always http_req (HTTP Request).

Possible values: http_req

Example

```
add transform policylabel trans_policylabel http_req
```

rm transform policylabel

Removes a URL Transformation policy label.

Synopsys

```
rm transform policylabel <labelName>
```

Arguments

labelName

Name of the policy label to remove.

Example

```
rm transform policylabel trans_policylabel
```

bind transform policylabel

Binds the specified URL Transformation policy to the specified policy label.

Synopsys

bind transform polycylabel <labelName> <policyName> <priority> [<gotoPriorityExpression>] [-invoke (<labelType> <labelName>)]

Arguments

labelName

Name of the policy label to be invoke if the current policy evaluates to TRUE, the invoke parameter is set, and the label type is Policy Label.

policyName

Name of the URL Transformation policy to bind to the policy label.

priority

Positive integer specifying the priority of the policy within the policy label. A lower number specifies a higher priority. Must be unique within the list of policies bound to the label. Policies are evaluated in the order of their priority numbers, and the first policy that matches is applied.

Minimum value: 1

Maximum value: 2147483647

gotoPriorityExpression

Optional expression or other value specifying the next policy to evaluate if the current policy evaluates to TRUE. Specify one of the following values:

- * NEXT ? Evaluate the policy with the next higher priority number.
- * END ? End policy evaluation.
- * USE_INVOCATION_RESULT ? Applicable if this policy invokes another policy label. If the final goto in the invoked policy label has a value of END, the evaluation stops. If the final goto is anything other than END, the current policy label performs a NEXT.
- * A PCRE-compatible regular expression that evaluates to a number.

If you specify an expression, the number to which it evaluates determines the next policy to evaluate, as follows:

- * If the expression evaluates to a higher numbered priority, the policy with that priority is evaluated next.
- * If the expression evaluates to the priority of the current policy, the policy with the next higher priority number is evaluated next.
- * If the expression evaluates to a number that is larger than the largest priority number, policy evaluation ends.

An UNDEF event is triggered if:

- * The expression is invalid.
- * The expression evaluates to a priority number that is smaller than the current policy's priority number.
- * The expression evaluates to a priority number that is between the current policy's priority number (say, 30) and the highest priority number (say, 100), but does not match any configured priority number (for example, the expression evaluates to the number 85). This example assumes that the priority number increments by 10 for every successive policy, and therefore a priority number of 85 does not exist in the policy label.

invoke

If the current policy evaluates to TRUE, terminate evaluation of policies bound to the current policy label, and then forward the request to the specified virtual server or evaluate the specified policy label.

labelType

Type of invocation. Available settings function as follows:

- * reqvserver - Forward the request to the specified request virtual server.

- * policylabel - Invoke the specified policy label.

Possible values: reqvserver, policylabel

Example

```
i) bind transform policylabel trans_policylabel pol_1 1 2 -invoke reqvserver CURRENT ii
```

unbind transform policylabel

Unbinds the specified URL Transformation policy from the specified policy label.

Synopsys

```
unbind transform policylabel <labelName> <policyName> [-priority <positive_integer>]
```

Arguments

labelName

Name of the label from which to unbind the policy.

policyName

Name of the label to which to bind the policy.

priority

Priority of the NOPOLICY to be unbound.

Minimum value: 1

Maximum value: 2147483647

Example

```
unbind transform policylabel trans_policylabel pol_1
```

show transform policylabel

Displays the current settings for the specified URL Transformation policy label. If no policy label is specified, displays a list of all URL Transformation policy labels currently configured on the NetScaler appliance.

Synopsys

```
show transform policylabel [<labelName>]
```

Arguments

labelName

Name for the policy label. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.) pound (#), space (), at (@), equals (=), colon (:), and underscore characters. Can be changed after the URL Transformation policy label is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my transform policylabel? or ?my transform policylabel).

Outputs

stateflag

policylabeltype

Types of transformations allowed by the policies bound to the label. For URL transformation, always http_req (HTTP Request).

numpol

Number of policies bound to label.

hits

Number of times policy label was invoked.

policyName

Name of the URL Transformation policy to bind to the policy label.

priority

Specifies the priority of the policy.

gotoPriorityExpression

Expression specifying the priority of the next policy which will get evaluated if the current policy rule evaluates to TRUE.

invoke

If the current policy evaluates to TRUE, terminate evaluation of policies bound to the current policy label, and then forward the request to the specified virtual server or evaluate the specified policy label.

labelType

Type of invocation. Available settings function as follows:

- * reqvserver - Forward the request to the specified request virtual server.
- * policylabel - Invoke the specified policy label.

labelName

Name of the policy label.

description

Description of the policylabel

flags

devno

count

Example

```
i) show transform policylabel trans_policylabel ii) show transform policylabel
```

stat transform policylabel

Displays statistics for the specified URL Transformation policy label. If no policy label name is provided, displays abbreviated statistics for all URL Transformation policy labels currently configured on the NetScaler appliance.

Synopsys

```
stat transform policylabel [<labelName>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>]  
[-clearstats ( basic | full )]
```


Arguments

labelName

The name of the URL Transformation policy label.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

Policy Label Hits (Hits)

Number of times policy label was invoked.

rename transform policylabel

Renames a URL Transformation policy label.

Synopsys

```
rename transform policylabel <labelName>@ <newName>@
```

Arguments

labelName

Current name of the policy label.

newName

New name for the policy label.

Must begin with a letter, number, or the underscore character (`_`), and must contain only letters, numbers, and the hyphen (`-`), period (`.`), pound (`#`), space (), at (`@`), equals (`=`), colon (`:`), and underscore characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, `?my transform policylabel?` or `?my transform policylabel`).

Example

```
rename transform policylabel oldname newname
```

transform profile

The following operations can be performed on "transform profile":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add transform profile

Creates a URL transformation profile, which contains a list of actions that define how the URLs in a request or response are to be modified. NOTE: In the URL Transformation feature (unlike all other NetScaler features), ?profile? and ?action? are not synonymous but refer to distinct entities. You must create the profile first, and then the actions.

Synopsys

add transform profile <name> [-type URL]

Arguments

name

Name for the URL transformation profile. Must begin with a letter, number, or the underscore character (_), and must contain only letters, numbers, and the hyphen (-), period (.), pound (#), space (), at (@), equals (=), colon (:), and underscore characters. Cannot be changed after the URL transformation profile is added.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, ?my transform profile? or ?my transform profile?).

type

Type of transformation. Always URL for URL Transformation profiles.

Possible values: URL

rm transform profile

Removes a URL Transformation profile.

Synopsys

rm transform profile <name>

Arguments

name

Name of the profile to remove.

set transform profile

Modifies the settings of a URL Transformation profile.

Synopsys

set transform profile <name> [-type URL] [-onlyTransformAbsURLinBody (ON | OFF)] [-comment <string>]

Arguments

name

Name of the profile to be modified.

type

Type of transformation. Always URL for URL Transformation profiles.

Possible values: URL

onlyTransformAbsURLinBody

In the HTTP body, transform only absolute URLs. Relative URLs are ignored.

Possible values: ON, OFF

comment

Any comments to preserve information about this URL Transformation profile.

unset transform profile

Use this command to remove transform profile settings. Refer to the set transform profile command for meanings of the arguments.

Synopsys

unset transform profile <name> [-type] [-onlyTransformAbsURLinBody] [-comment]

show transform profile

Displays the current settings for the specified URL Transformation profile. If no URL Transformation profile name is specified, displays a list of all URL Transformation profiles currently configured on the NetScaler appliance.

Synopsys

show transform profile [<name>]

Arguments

name

Name of the profile.

Outputs

actionName

URL Transformation action name.

stateflag

type

Type of transformation. Always URL for URL Transformation profiles.

RegexForFindingURLinJavaScript

Patclass having regexes to find the URLs in JavaScript.

RegexForFindingURLinCSS

Patclass having regexes to find the URLs in CSS.

RegexForFindingURLinXComponent

Patclass having regexes to find the URLs in X-Component.

RegexForFindingURLinXML

Patclass having regexes to find the URLs in XML.

additionalReqHeadersList

Patclass having a list of additional request header names that should transformed.

additionalRespHeadersList

Patclass having a list of additional response header names that should transformed.

onlyTransformAbsURLinBody

In the HTTP body, transform only absolute URLs. Relative URLs are ignored.

comment

Any comments to preserve information about this URL Transformation profile.

priority

Priority of the Action within the Profile.

state

Enabled flag.

profileName

URL Transformation profile name.

reqUrlFrom

Pattern of original request URLs. It corresponds to the way external users view the server, and acts as a source for request transformations.

reqUrlInto

Pattern of transformed request URLs. It corresponds to internal addresses and indicates how they are created.

resUrlFrom

Pattern of original response URLs. It corresponds to the way external users view the server, and acts as a source for response transformations.

resUrlInto

Pattern of transformed response URLs. It corresponds to internal addresses and indicates how they are created.

cookieDomainFrom

Pattern of the original domain in Set-Cookie headers.

cookieDomainInto

Pattern of the transformed domain in Set-Cookie headers.

actionComment

Comments.

devno

count

Tunnel Commands

The entities on which you can perform NetScaler CLI operations:

- o tunnel global
- o tunnel trafficPolicy

tunnel global

The following operations can be performed on "tunnel global":

[bind](#) | [unbind](#) | [show](#)

bind tunnel global

Activates an existing tunnel traffic policy globally.

Synopsys

```
bind tunnel global (<policyName> [-priority <positive_integer>]) [-state ( ENABLED | DISABLED )]
```

Arguments

policyName

Name of the tunnel traffic policy to activate or bind.

priority

Integer specifying the policy's priority. The lower the number, the higher the priority. Policies are evaluated in the order of their priority numbers.

Minimum value: 0

Maximum value: 64000

state

Current state of the binding. If the binding is enabled, the policy is active.

Possible values: ENABLED, DISABLED

Default value: ENABLED

Example

```
add tunnel trafficpolicy cmp_all_destport "REQ.TCP.DESTPORT == 0-65535" GZIP After crea
```

unbind tunnel global

Deactivates an active tunnel traffic policy.

Synopsys

```
unbind tunnel global <policyName>
```

Arguments

policyName

Name of the tunnel traffic policy to unbind or deactivate.

Example

```
Globally active tunnel policies can be seen using command: > show tunnel global 1 Glob
```

show tunnel global

Displays globally active tunnel policies.

Synopsys

show tunnel global

Outputs

policyName

Policy name.

priority

Priority.

state

Current state of the binding. If the binding is enabled, the policy is active.

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

stateflag

devno

count

Example

```
> sh tunnel global 1) Policy Name: cmp_all_destport   Priority: 0 2) Policy Name: local_;
```


tunnel trafficPolicy

The following operations can be performed on "tunnel trafficPolicy":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add tunnel trafficPolicy

Creates a tunnel traffic policy. A tunnel traffic policy defines the type of compression to be used for the tunneled traffic.

Synopsys

add tunnel trafficPolicy <name> <rule> <action>

Arguments

name

Name for the tunnel traffic policy.

Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my policy" or 'my policy').

rule

Expression, against which traffic is evaluated. Written in classic or default syntax.

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters> + <string of 245 characters>"

The following requirements apply only to the NetScaler CLI:

- * If the expression includes blank spaces, the entire expression must be enclosed in double quotation marks.
- * If the expression itself includes double quotation marks, you must escape the quotations by using the \ character.
- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Name of the built-in compression action to associate with the policy.

Example

Example 1: add tunnel trafficpolicy cmp_all_destport "REQ.TCP.DESTPORT == 0-65535" GZIP

rm tunnel trafficPolicy

Removes a tunnel traffic policy.

Synopsys

rm tunnel trafficPolicy <name>

Arguments

name

Name of the tunnel traffic policy to remove.

Example

```
rm tunnel trafficpolicy tunnel_policy_name
```

 The "show tunnel trafficpolicy" command shows :

set tunnel trafficPolicy

Modifies the specified parameters of an existing tunnel traffic policy.

Synopsys

```
set tunnel trafficPolicy <name> [-rule <expression>] [-action <string>]
```

Arguments

name

Name of the tunnel traffic policy to modify.

rule

Expression, against which traffic is evaluated. Written in classic or default syntax.

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

The following requirements apply only to the NetScaler CLI:

- * If the expression includes blank spaces, the entire expression must be enclosed in double quotation marks.
- * If the expression itself includes double quotation marks, you must escape the quotations by using the \ character.
- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Name of the built-in compression action to associate with the policy.

Example

```
add tunnel trafficpolicy cmp_all_destport "REQ.TCP.DESTPORT == 0-65535" GZIP set tunnel trafficPolicy cmp_all_destport
```

unset tunnel trafficPolicy

Use this command to remove tunnel trafficPolicy settings. Refer to the set tunnel trafficPolicy command for meanings of the arguments.

Synopsys

```
unset tunnel trafficPolicy <name> [-rule] [-action]
```

show tunnel trafficPolicy

Displays information about all the configured tunnel traffic policies, or displays detailed information about the specified tunnel traffic policy.

Synopsys

show tunnel trafficPolicy [<name>]

Arguments

name

Name of the tunnel traffic policy for which to show detailed information.

Outputs

rule

Expression, against which traffic is evaluated. Written in classic or default syntax.

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

The following requirements apply only to the NetScaler CLI:

- * If the expression includes blank spaces, the entire expression must be enclosed in double quotation marks.
- * If the expression itself includes double quotation marks, you must escape the quotations by using the \ character.
- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Name of the built-in compression action to associate with the policy.

hits

No of hits.

txbytes

Number of bytes transmitted.

rxbytes

Number of bytes received.

clientTTLB

Total client TTLB value.

clientTransactions

Number of client transactions.

serverTTLB

Total server TTLB value.

serverTransactions

Number of server transactions.

boundTo

The entity name to which policy is bound

activePolicy

priority

flags

bindPolicyType

isDefault

A value of true is returned if it is a default tunnelpolicy.

policyType

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

devno

count

stateflag

Example

```
> show tunnel trafficpolicy          2 Tunnel policies: 1) Name: local_sub_nocmp   Rule: ;
```

Utility Commands

The entities on which you can perform NetScaler CLI operations:

- o audit
- o callhome
- o grep
- o install
- o nstrace
- o ping
- o ping6
- o raid
- o scp
- o shell
- o techsupport
- o traceroute
- o traceroute6

audit

The following operations can be performed on "audit":

config audit

audit and verify the commands in file against running config. NOTE: This command is deprecated. Command deprecated. Use diff ns config command

Synopsys

Arguments

commandStr

specify the options.

Example

```
config audit -diff -f <filename>
```

callhome

The following operations can be performed on "callhome":

[show](#) | [set](#) | [unset](#)

show callhome

Displays the trigger events configured and the time when these events were triggered.

Synopsys

show callhome

Outputs

emailAddress

The contact person's E-mail address.

proxyMode

Deploy the callhome proxy mode

IPAddress

Proxy Server IP address

port

Proxy Server Port

sslcardfirstfailure

First occurrence SSL card failure.

sslcardlatestfailure

Latest occurrence SSL card failure.

powfirstfail

First occurrence power supply unit failure.

powlatestfailure

Latest occurrence power supply unit failure.

HDDfirstfail

First occurrence hard disk drive failure.

HDDlatestfailure

Latest occurrence hard disk drive failure.

FLASHfirstfail

First occurrence compact flash failure.

FLASHlatestfailure

Latest occurrence compact flush failure.

restartLatestfail

Latest occurrence warm restart failure.

callhomestatus

Callhome feature enabled/disable, register with upload server successful/failed

Example

```
show callhome E-mail address configured:xxx@yahoo.com Trigger event State
```

set callhome

Sets the contact person's E-mail address

Synopsys

```
set callhome -emailAddress e-mailaddress
```

Arguments

emailAddress

The contact person's E-mail address.

proxyMode

Deploy the callhome proxy mode

Possible values: YES, NO

Default value: NO

IPAddress

Proxy Server IP address

port

Proxy Server Port

Minimum value: 1

Example

```
set callhome -emailAddress xxxx@yahoo.com
```

unset callhome

Use this command to remove callhome settings.Refer to the set callhome command for meanings of the arguments.

Synopsys

```
unset callhome [-emailAddress] [-proxyMode] [-IPAddress] [-port]
```


grep

The following operations can be performed on "grep":

grep

Searches files or output for lines containing a match to the specified <pattern>. By default, grep prints the matching lines.

Synopsis

```
grep [-c] [-E] [-i] [-v] [-w] [-x] <pattern>
```

Arguments

c

Suppress normal output. Instead print a count of matching lines.

With the -v option, count non-matching lines.

E

Interpret <pattern> as an extended regular expression.

i

Ignore case distinctions.

v

Invert the sense of matching, to select non-matching lines.

w

Select only those lines containing matches that form whole words.

x

Select only those matches that exactly match the whole line.

pattern

The pattern (regular expression or text string) for which to search.

Example

```
show ns info | grep off -i
```

install

The following operations can be performed on "install":

install

Installs a version of NetScaler software on the system.

Synopsys

```
install <url> [-c] [-y]
```

Arguments

url

http://[user]:[password]@host/path/to/file

https://[user]:[password]@host/path/to/file

sftp://[user]:[password]@host/path/to/file

scp://[user]:[password]@host/path/to/file

ftp://[user]:[password]@host/path/to/file

file://path/to/file

c

Back up existing kernel.

y

Do not prompt for yes/no before rebooting.

Example

```
install http://host.netscaler.com/ns-6.0-41.2.tgz
```

nstrace

The following operations can be performed on "nstrace":

nstrace

Invokes the nstrace program to log traffic flowing through the NetScaler appliance.

Synopsys

```
nstrace [-nf <positive_integer>] [-time <secs>] [-size <positive_integer>] [-mode <mode> ...] [-tcpdump ( ENABLED | DISABLED )] [-perNIC ( ENABLED | DISABLED )]] [-name <string> [-id <string>]] [-filter <expression> [-link ( ENABLED | DISABLED )]]
```

Arguments

nf

Number of files to be generated in a single run of the command.

Default value: 24

Minimum value: 0

time

Number of seconds for which to log to trace file. Can be a mathematical expression. For example, to log to trace files for 2 hours, you can specify 2*60*60.

Default value: 3600

size

Size of the packet to be logged (should be in the range of 60 to 1514 bytes). Set to 0 for full packet trace.

Default value: 164

Minimum value: 0

Maximum value: 1514

mode

Capturing mode for trace. Can be any of the following values, or a combination of these values:

- * RX - Received packets before NIC pipelining
- * NEW_RX - Received packets after NIC pipelining (packets that are not dropped)
- * TX - Transmitted packets
- * TXB - Packets buffered for transmission
- * IPV6 - Translated IPv6 packets
- * C2C - Capture core-to-core messages
- * NS_FR_TX - Flow receiver does not capture the TX/TXB packets. Applicable only for a cluster setup.

You can also provide a combination of modes. For example:

- * -mode NEW_RX TXB: Capture RX packets after NIC handling and packets that are buffered for actual transmission.
- * -mode RX TX: Capture packet during NIC pipeline (filter expressions will not work for RX mode).
- * -mode NEW_RX TXB NS_FR_TX: Default mode except that TX/TXB packets on the flow receiver are not captured.

Default value: DEFAULT_MODE

tcpdump

Log files format supported:nstrace-format, tcpdump-format. default:nstrace-format

Possible values: ENABLED, DISABLED

Default value: DISABLED

perNIC

Use separate trace files for each interface. Works only with TCP dump format.

Possible values: ENABLED, DISABLED

Default value: DISABLED

name

Custom file name for nstrace files.

id

ID for the trace file name, for uniqueness. Use only with the -name option.

filter

Filter expression for nstrace. Maximum length of filter is 255 and it can be of the following format:

"<expression> [<relop> <expression>"]

where,

<relop> can be the && or the || relational operators.

<expression> is a string in the following format: <qualifier> <operator> <qualifier-value>

where,

<operator> can be any one of the following (except the commas): ==, eq, !=, neq, >, gt, <, lt, >=, ge, <=, le, BETWEEN

Following are the valid qualifiers for the command: SOURCEIP, SOURCEPORT, DESTIP, DESTPORT, IP, PORT, SVCNAME, VSVRNAME, CONNID, VLAN, INTF.

Example:

nstrace -filter "SOURCEIP==10.102.34.201 || SVCNAME !=s1 && SOURCEPORT >80"

link

Include peer traffic of filtered connections.

Possible values: ENABLED, DISABLED

Default value: DISABLED

Example

```
nstrace -nf 10 -time 100 -mode RX IPV6 TXB -name abc -tcpdump ENABLED -perNIC ENABLED
```

ping

The following operations can be performed on "ping":

ping

Invokes the UNIX ping command. The hostName parameter must be used if the name is in the /etc/hosts file directory or is otherwise known in DNS.

Synopsys

```
ping [-c <count>] [-i <interval>] [-I <interface>] [-n] [-p <pattern>] [-q] [-s <size>] [-S <src_addr>] [-T <td>] [-t <timeout>] <hostname>
```

Arguments

c

Number of packets to send. The default value is infinite.

Minimum value: 1

Maximum value: 65535

i

Waiting time, in seconds. The default value is 1 second.

Minimum value: 0

Maximum value: 65535

I

Network interface on which to ping, if you have multiple interfaces.

n

Numeric output only. No name resolution.

p

Pattern to fill in packets. Can be up to 16 bytes, useful for diagnosing data-dependent problems.

q

Quiet output. Only the summary is printed.

s

Data size, in bytes. The default value is 56.

Minimum value: 0

Maximum value: 65507

S

Source IP address to be used in the outgoing query packets. If the IP address does not belong to this appliance, an error is returned and nothing is sent.

T

Traffic Domain Id

Minimum value: 1

Maximum value: 4094

t

Time-out, in seconds, before ping exits.

Minimum value: 1

Maximum value: 3600

hostName

Address of host to ping.

Example

```
ping -p ff -c 4 10.102.4.107
```

ping6

The following operations can be performed on "ping6":

ping6

Invokes the UNIX ping6 command. The hostName parameter must be used if the name is in the /etc/hosts file directory or is otherwise known in DNS.

Synopsys

```
ping6 [-b <bufsiz>] [-c <count>] [-i <interval>] [-I <interface>] [-m] [-n] [-p <pattern>] [-q] [-S sourceaddr] [-V <vlanid>] [-T <td>] [-s <size>] Hostname
```

Arguments

b

Set socket buffer size. If used, should be used with roughly +100 then the datalen (-s option). The default value is 8192.

Minimum value: 132

Maximum value: 131071

c

Number of packets to send. The default value is infinite.

Minimum value: 1

Maximum value: 65535

i

Waiting time, in seconds. The default value is 1 second.

Minimum value: 0

Maximum value: 65535

I

Network interface on which to ping, if you have multiple interfaces.

m

By default, ping6 asks the kernel to fragment packets to fit into the minimum IPv6 MTU. The -m option will suppress the behavior for unicast packets.

n

Numeric output only. No name resolution.

p

Pattern to fill in packets. Can be up to 16 bytes, useful for diagnosing data-dependent problems.

q

Quiet output. Only summary is printed.

s

Data size, in bytes. The default value is 32.

Minimum value: 0

Maximum value: 65527

V

VLAN ID for link local address.

Minimum value: 1

Maximum value: 4094

S

Source IP address to be used in the outgoing query packets.

T

Traffic Domain Id

Minimum value: 1

Maximum value: 4094

hostName

Address of host to ping.

Example

```
ping6 -p ff -I 1/1 -c 4 2002::1
```


raid

The following operations can be performed on "raid":

show raid

Provides status of raid

Synopsys

show raid

Example

```
status raid
```

scp

The following operations can be performed on "scp":

scp

Securely copies data from one computer to another, in SSH protocol.

Synopsys

```
scp [-r] [-C] [-q] <sourceString> <destString>
```

Arguments

r

Recursively copy subdirectories.

C

Enable compression.

q

Quiet output. Disable the progress meter.

sourceString

Source user, host, and file path, specified as <user>@<host>:<path_to_copy_from>. The user and host parts are optional.

destString

Destination user, host, and file path, specified as

<user>@<host>:<path_to_copy_to>. The user and host parts are optional.

Example

```
scp /nsconfig/ns.conf nsroot@10.102.4.107:/nsconfig/
```

shell

The following operations can be performed on "shell":

shell

Exits to the FreeBSD command prompt. Press Control + D or type exit to return to the NetScaler command prompt.
Note: The shell can be accessed only by users who have write access to the NetScaler appliance.

Synopsys

shell [(command)]

Arguments

command

Shell command(s) to be invoked.

Example

```
> shell # ps | grep nscli 485  p0  S      0:01.12 -nscli (nscli) 590  p0  S+      0:00.00 9
```

techsupport

The following operations can be performed on "techsupport":

show techsupport

Generates a tar of system configuration data and statistics. This file must be submitted to Citrix technical support with file name collector_<NS IP>_<P/S>_<DateTime>.tgz. The archive is always pointed by the symbolic link /var/tmp/support/support.tgz for each invocation of the command.

Synopsys

```
show techsupport [-scope ( NODE | CLUSTER )]
```

Arguments

scope

Use this option to run showtechsupport on present node or all cluster nodes

Possible values: NODE, CLUSTER

Default value: NODE

Outputs

response

response as a text blob

serverName

Example

```
show techsupport
```

traceroute

The following operations can be performed on "traceroute":

traceroute

Invokes the UNIX traceroute command. This command attempts to track the route that the packets follow to reach the destination host.

Synopsys

```
traceroute [-S] [-n] [-r] [-v] [-M <min_ttl>] [-m <max_ttl>] [-P <protocol>][ -p <portno>] [-q <nqueries>] [-s <src_addr>] [-T <td>] [-t <tos>] [-w <wait>] <host> [<packetlen>]
```

Arguments

s

Print a summary of how many probes were not answered for each hop.

n

Print hop addresses numerically instead of symbolically and numerically.

r

Bypass normal routing tables and send directly to a host on an attached network. If the host is not on a directly attached network, an error is returned.

v

Verbose output. List received ICMP packets other than TIME_EXCEEDED and UNREACHABLE.

M

Minimum TTL value used in outgoing probe packets.

Default value: 1

Minimum value: 1

Maximum value: 255

m

Maximum TTL value used in outgoing probe packets.

Default value: 64

Minimum value: 1

Maximum value: 255

P

Send packets of specified IP protocol. The currently supported protocols are UDP and ICMP.

p

Base port number used in probes.

Default value: 33434

Minimum value: 1

Maximum value: 65535

q

Number of queries per hop.

Default value: 3

Minimum value: 1

Maximum value: 65535

s

Source IP address to use in the outgoing query packets. If the IP address does not belong to this appliance, an error is returned and nothing is sent.

T

Traffic Domain Id

Minimum value: 1

Maximum value: 4094

t

Type-of-service in query packets.

Maximum value: 255

w

Time (in seconds) to wait for a response to a query.

Default value: 5

Minimum value: 2

Maximum value: 86399

host

Destination host IP address or name.

packetlen

Length (in bytes) of the query packets.

Default value: 44

Minimum value: 44

Maximum value: 32768

Example

```
traceroute 10.102.4.107
```

traceroute6

The following operations can be performed on "traceroute6":

traceroute6

Invokes the UNIX traceroute6 command. Traceroute6 attempts to track the route that the packets follow to reach the destination host.

Synopsis

```
traceroute6 [-n] [I] [-r] [-v] [-m <hoplimit>] [-p <port>] [-q <probes>] [-s <src_addr>] [-T <td>] [-w <waittime>] <target> [<packetlen>]
```

Arguments

n

Print hop addresses numerically rather than symbolically and numerically.

I

Use ICMP ECHO for probes.

r

Bypass normal routing tables and send directly to a host on an attached network. If the host is not on a directly attached network, an error is returned.

v

Verbose output. List received ICMP packets other than TIME_EXCEEDED and UNREACHABLE.

m

Maximum hop value for outgoing probe packets.

Default value: 64

Minimum value: 1

Maximum value: 255

p

Base port number used in probes.

Default value: 33434

Minimum value: 1

Maximum value: 65535

q

Number of probes per hop.

Default value: 3

Minimum value: 1

Maximum value: 65535

s

Source IP address to use in the outgoing query packets. If the IP address does not belong to this appliance, an error is returned and nothing is sent.

T

Traffic Domain Id

Minimum value: 1

Maximum value: 4094

w

Time (in seconds) to wait for a response to a query.

Default value: 5

Minimum value: 2

Maximum value: 86399

host

Destination host IP address or name.

packetlen

Length (in bytes) of the query packets.

Default value: 44

Minimum value: 44

Maximum value: 32768

Example

```
traceroute6 2002::7
```


VPN Commands

The entities on which you can perform NetScaler CLI operations:

- o vpn
- o vpn clientlessAccessPolicy
- o vpn clientlessAccessProfile
- o vpn epaprofile
- o vpn formSSOAction
- o vpn global
- o vpn icaConnection
- o vpn intranetApplication
- o vpn nextHopServer
- o vpn parameter
- o vpn samlSSOProfile
- o vpn sessionAction
- o vpn sessionPolicy
- o vpn stats
- o vpn trafficAction
- o vpn trafficPolicy
- o vpn url
- o vpn vserver

vpn

The following operations can be performed on "vpn":

stat vpn

Displays the statistics for NetScaler Gateway usage. Displays event information, such as the event that generated the message, a time stamp, the message type, and predefined log levels and message information.

Synopsys

```
stat vpn [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats ( basic | full )]
```

Arguments

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

Login-page requests received (iHtHit)

Number of requests for VPN login page.

Login-page delivery failures (iHtFail)

Number of failures to display VPN login page.

Client-configuration requests (cfgHit)

Number of client configuration requests received by VPN server.

DNS queries resolved (dnsHit)

Number of DNS queries resolved by VPN server.

WINS queries resolved (winsHit)

Number of WINS queries resolved by VPN server.

Number of SSLVPN tunnels (csHit)

Number of SSL VPN tunnels formed between VPN server and client.

Backend non-HTTP server probes (csNoHttp)

Number of probes from VPN to back-end non-HTTP servers that have been accessed by the VPN client.

Backend HTTP server probes (csHttp)

Number of probes from VPN to back-end HTTP servers that have been accessed by the VPN client.

Backend server probe successes (csConSuc)

Number of successful probes to all back-end servers.

File-system requests received (totFsHit)

Number of file system requests received by VPN server.

IIP disabled and MIP used (IIPdMIPu)

Number of times MIP is used as IIP is disabled.

IIP failed and MIP used (IIPfMIPu)

Number of times MIP is used as IIP assignment failed.

IIP spillover and MIP used (IIPsMIPu)

Number of times MIP is used on IIP Spillover.

IIP disabled and MIP disabled (IIPdMIPd)

Both IIP and MIP is disabled.

IIP failed and MIP disabled (IIPfMIPd)

Number of times IIP assignment failed and MIP is disabled.

SOCKS method request received (SOCKSmReqR)

Number of received SOCKS method request.

SOCKS method request sent (SOCKSmReqS)

Number of sent SOCKS method request.

SOCKS method response received (SOCKSmRespR)

Number of received SOCKS method response.

SOCKS method response sent (SOCKSmRespS)

Number of sent SOCKS method response.

SOCKS connect request received (SOCKScReqR)

Number of received SOCKS connect request.

SOCKS connect request sent (SOCKScReqS)

Number of sent SOCKS connect request.

SOCKS connect response received (SOCKScRespR)

Number of received SOCKS connect response.

SOCKS connect response sent (SOCKScRespS)

Number of sent SOCKS connect response.

SOCKS server error (SOCKSserverErr)

Number of SOCKS server error.

SOCKS client error (SOCKSclientErr)

Number of SOCKS client error.

STA connection success (STAconnSucc)

Number of STA connection success.

STA connection failure (STAconnFail)

Number of STA connection failure.

CPS connection success (CPSconnSucc)

Number of CPS connection success.

CPS connection failure (CPSconnFail)

Number of CPS connection failure.

STA request sent (STAreqSent)

Number of STA request sent.

STA response received (STArespRecvd)

Number of STA response received.

ICA license failure (ICALicenseFail)

Number of ICA license failure.

vpn clientlessAccessPolicy

The following operations can be performed on "vpn clientlessAccessPolicy":

[add](#) | [rm](#) | [set](#) | [show](#)

add vpn clientlessAccessPolicy

Adds a clientless access policy, which enables users to log on using a web browser and connect to the bookmarked web address without requiring the user to install a software plug-in.

Synopsys

add vpn clientlessAccessPolicy <name> <rule> <profileName>

Arguments

name

Name of the new clientless access policy.

rule

Expression, or name of a named expression, specifying the traffic that matches the policy. Can be written in either default or classic syntax.

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

The following requirements apply only to the NetScaler CLI:

- * If the expression includes one or more spaces, enclose the entire expression in double quotation marks.
- * If the expression itself includes double quotation marks, escape the quotations by using the \ character.
- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

profileName

Name of the profile to invoke for the clientless access.

rm vpn clientlessAccessPolicy

Removes a clientless access policy.

Synopsys

rm vpn clientlessAccessPolicy <name>

Arguments

name

Name of the clientless access policy to remove.

set vpn clientlessAccessPolicy

Adds a new rule to be used by an existing clientless access policy that includes a simple expression that specifies the conditions for which the policy is enforced.

Synopsys

set vpn clientlessAccessPolicy <name> [-rule <expression>] [-profileName <string>]

Arguments

name

Name of the existing clientless access policy to modify.

rule

Expression, or name of a named expression, specifying the traffic that matches the policy. Can be written in either default or classic syntax.

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

The following requirements apply only to the NetScaler CLI:

- * If the expression includes one or more spaces, enclose the entire expression in double quotation marks.
- * If the expression itself includes double quotation marks, escape the quotations by using the \ character.
- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

profileName

Name of the profile to invoke for the clientless access.

show vpn clientlessAccessPolicy

Displays a clientless access policy.

Synopsys

show vpn clientlessAccessPolicy [<name>]

Arguments

name

Name of the clientless access policy to display.

Outputs

rule

The rule used by the clientless access policy. Rules are combinations of expressions. Expressions are simple conditions, such as a test for equality, applied to operands, such as a URL string or an IP address. Expression syntax is described in the Installation and Configuration Guide

profileName

The profile to invoked for the clientless access.

undefAction

The UNDEF action.

hits

The number of times the policy evaluated to true.

undefHits

The number of times the policy evaluation resulted in undefined processing.

activePolicy

Indicates whether policy is bound or not.

boundTo

Location where policy is bound.

priority

Specifies the priority of the policy.

description

Description of the clientless access policy.

isDefault

A value of true is returned if it is a default vpnclientlessrwpolicy.

stateflag**builtin**

Flag to determine if vpn clientless rewrite policy is built-in or not

devno**count**

vpn clientlessAccessProfile

The following operations can be performed on "vpn clientlessAccessProfile":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add vpn clientlessAccessProfile

Adds a collection of settings that allows clientless access to a given application. Settings include the policies to specify whether to rewrite a URL, rules to find the URLs within various web content-types, and a set of cookies that are required to be present on the client machine.

Synopsys

add vpn clientlessAccessProfile <profileName>

Arguments

profileName

Name for the NetScaler Gateway clientless access profile. Must begin with an ASCII alphabetic or underscore (_) character, and must consist only of ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the profile is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my profile" or 'my profile').

rm vpn clientlessAccessProfile

Removes a clientless access profile.

Synopsys

rm vpn clientlessAccessProfile <profileName>

Arguments

profileName

Name of the clientless access profile to remove.

set vpn clientlessAccessProfile

Modifies the settings for an existing clientless access profile.

Synopsys

set vpn clientlessAccessProfile <profileName> [-URLRewritePolicyLabel <string>] [-JavaScriptRewritePolicyLabel <string>] [-ReqHdrRewritePolicyLabel <string>] [-ResHdrRewritePolicyLabel <string>] [-RegexForFindingURLinJavaScript <string>] [-RegexForFindingURLinCSS <string>] [-RegexForFindingURLinXComponent <string>] [-RegexForFindingURLinXML <string>] [-RegexForFindingCustomURLs <string>] [-ClientConsumedCookies <string>] [-requirePersistentCookie (ON | OFF)]

Arguments

profileName

Name of the clientless access profile to modify.

URLRewritePolicyLabel

Name of the configured URL rewrite policy label. If you do not specify a policy label name, then URLs are not rewritten.

JavaScriptRewritePolicyLabel

Name of the configured JavaScript rewrite policy label. If you do not specify a policy label name, then JAVA scripts are not rewritten.

ReqHdrRewritePolicyLabel

Name of the configured Request rewrite policy label. If you do not specify a policy label name, then requests are not rewritten.

ResHdrRewritePolicyLabel

Name of the configured Response rewrite policy label.

RegexForFindingURLinJavaScript

Name of the pattern set that contains the regular expressions, which match the URL in Java script.

RegexForFindingURLinCSS

Name of the pattern set that contains the regular expressions, which match the URL in the CSS.

RegexForFindingURLinXComponent

Name of the pattern set that contains the regular expressions, which match the URL in X Component.

RegexForFindingURLinXML

Name of the pattern set that contains the regular expressions, which match the URL in XML.

RegexForFindingCustomURLs

Name of the pattern set that contains the regular expressions, which match the URLs in the custom content type other than HTML, CSS, XML, XCOMP, and JavaScript. The custom content type should be included in the patset ns_cvpn_custom_content_types.

ClientConsumedCookies

Specify the name of the pattern set containing the names of the cookies, which are allowed between the client and the server. If a pattern set is not specified, NetScaler Gateway does not allow any cookies between the client and the server. A cookie that is not specified in the pattern set is handled by NetScaler Gateway on behalf of the client.

requirePersistentCookie

Specify whether a persistent session cookie is set and accepted for clientless access. If this parameter is set to ON, COM objects, such as MSOffice, which are invoked by the browser can access the files using clientless access. Use caution because the persistent cookie is stored on the disk.

Possible values: ON, OFF

Default value: OFF

unset vpn clientlessAccessProfile

Resets the attributes of the specified clientless access profile. Attributes for which a default value is available revert to their default values. Refer to the set vpn clientlessAccessProfile command for a description of the parameters..Refer to the set vpn clientlessAccessProfile command for meanings of the arguments.

Synopsys

```
unset vpn clientlessAccessProfile <profileName> [-URLRewritePolicyLabel] [-JavaScriptRewritePolicyLabel] [-ReqHdrRewritePolicyLabel] [-ResHdrRewritePolicyLabel] [-RegexForFindingURLinJavaScript] [-RegexForFindingURLinCSS] [-RegexForFindingURLinXComponent] [-RegexForFindingURLinXML] [-RegexForFindingCustomURLs] [-ClientConsumedCookies] [-requirePersistentCookie]
```

show vpn clientlessAccessProfile

Displays information about all the configured clientless access profiles, or displays detailed information about the specified clientless access profile.

Synopsys

show vpn clientlessAccessProfile [<profileName>]

Arguments

profileName

Name of the clientless access profile for which to display detailed information.

Outputs

stateflag

URLRewritePolicyLabel

Name of the configured URL rewrite policy label. If you do not specify a policy label name, then URLs are not rewritten.

JavaScriptRewritePolicyLabel

Name of the configured JavaScript rewrite policy label. If you do not specify a policy label name, then JAVA scripts are not rewritten.

CSSRewritePolicyLabel

The configured CSS rewrite policylabel.

XMLRewritePolicyLabel

The configured XML rewrite policylabel.

XComponentRewritePolicyLabel

The configured X-Component rewrite policylabel.

ReqHdrRewritePolicyLabel

Name of the configured Request rewrite policy label. If you do not specify a policy label name, then requests are not rewritten.

ResHdrRewritePolicyLabel

Name of the configured Response rewrite policy label.

RegexForFindingURLinJavaScript

Name of the pattern set that contains the regular expressions, which match the URL in Java script.

RegexForFindingURLinCSS

Name of the pattern set that contains the regular expressions, which match the URL in the CSS.

RegexForFindingURLinXComponent

Name of the pattern set that contains the regular expressions, which match the URL in X Component.

RegexForFindingURLinXML

Name of the pattern set that contains the regular expressions, which match the URL in XML.

RegexForFindingCustomURLs

Name of the pattern set that contains the regular expressions, which match the URLs in the custom content type other than HTML, CSS, XML, XCOMP, and JavaScript. The custom content type should be included in the patset ns_cvpn_custom_content_types.

ClientConsumedCookies

Specify the name of the pattern set containing the names of the cookies, which are allowed between the client and the server. If a pattern set is not specified, NetScaler Gateway does not allow any cookies between the client and the server. A cookie that is not specified in the pattern set is handled by NetScaler Gateway on behalf of the client.

requirePersistentCookie

Specify whether a persistent session cookie is set and accepted for clientless access. If this parameter is set to ON, COM objects, such as MSOffice, which are invoked by the browser can access the files using clientless access. Use caution because the persistent cookie is stored on the disk.

isDefault

A value of true is returned if it is a default vpnclientlessrwprofile.

description

Description of the clientless access profile.

builtin

Flag to determine if vpn clientless rewrite profile is built-in or not

devno

count

vpn epaprofile

The following operations can be performed on "vpn epaprofile":

[add](#) | [rm](#) | [show](#)

add vpn epaprofile

Creates an advanced pre-authentication EPA device profile with XML data NOTE: This command is deprecated.
Deprecated AdvanceEPA Option

Synopsys

Arguments

name

name of device profile

fileName

filename of the deviceprofile data xml

data

deviceprofile data xml

rm vpn epaprofile

Removes a previously created EPA device profile. NOTE: This command is deprecated.Deprecated AdvanceEPA Option

Synopsys

Arguments

name

Name of EPA device profile to remove

show vpn epaprofile

Displays information on device profile NOTE: This command is deprecated.Deprecated AdvanceEPA Option

Synopsys

Arguments

name

name of device profile.

Outputs

data

The XML data associated with the device profile

devno

count

stateflag

Example

```
show vpn deviceprofile DP1
```

vpn formSSOAction

The following operations can be performed on "vpn formSSOAction":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add vpn formSSOAction

Creates a form-based single sign-on profile. Form based single sign-on allows users to log on one time to all protected applications in your network. Users can access web applications that require an HTML form-based logon without having to type their password again.

Synopsys

```
add vpn formSSOAction <name> -actionURL <URL> -userField <string> -passwdField <string> -ssoSuccessRule
<expression> [-nameValuePair <string>] [-responsesize <positive_integer>] [-nvtype ( STATIC | DYNAMIC )] [-
submitMethod ( GET | POST )]
```

Arguments

name

Name for the form based single sign-on profile.

actionURL

Root-relative URL to which the completed form is submitted.

userField

Name of the form field in which the user types in the user ID.

passwdField

Name of the form field in which the user types in the password.

ssoSuccessRule

Use a frequently used expression or create a custom expression describing the action that the form-based single sign-on profile takes when invoked by a policy. Used for verifying successful single sign-on.

nameValuePair

Other name-value pair attributes to send to the server, in addition to sending the user name and password. Value names are separated by an ampersand (&), such as in name1=value1&name2=value2.

responsesize

Maximum number of bytes to allow in the response size. Specifies the number of bytes in the response to be parsed for extracting the forms.

Default value: 8096

Minimum value: 0

nvtype

How to process the name-value pair. Available settings function as follows:

* STATIC - The administrator-configured values are used.

* DYNAMIC - The response is parsed, the form is extracted, and then submitted.

Possible values: STATIC, DYNAMIC

Default value: DYNAMIC

submitMethod

HTTP method (GET or POST) used by the single sign-on form to send the logon credentials to the logon server.

Possible values: GET, POST

Default value: GET

rm vpn formSSOAction

Removes a configured form-based single sign-on profile.

Synopsis

```
rm vpn formSSOAction <name>
```

Arguments

name

Name of the form-based single sign-on profile to remove.

set vpn formSSOAction

Modifies the parameters of an existing form-based single sign-on profile (or action).

Synopsis

```
set vpn formSSOAction <name> [-actionURL <URL>] [-userField <string>] [-passwdField <string>] [-ssoSuccessRule <expression>] [-responsesize <positive_integer>] [-nameValuePair <string>] [-nvtype ( STATIC | DYNAMIC )] [-submitMethod ( GET | POST )]
```

Arguments

name

Name for the form based single sign-on profile.

actionURL

Root-relative URL to which the completed form is submitted.

userField

Name of the form field in which the user types in the user ID.

passwdField

Name of the form field in which the user types in the password.

ssoSuccessRule

Use a frequently used expression or create a custom expression describing the action that the form-based single sign-on profile takes when invoked by a policy. Used for verifying successful single sign-on.

responsesize

Maximum number of bytes to allow in the response size. Specifies the number of bytes in the response to be parsed for extracting the forms.

Default value: 8096

Minimum value: 0

nameValuePair

Other name-value pair attributes to send to the server, in addition to sending the user name and password. Value names are separated by an ampersand (&), such as in name1=value1&name2=value2.

nvtype

How to process the name-value pair. Available settings function as follows:

- * STATIC - The administrator-configured values are used.
- * DYNAMIC - The response is parsed, the form is extracted, and then submitted.

Possible values: STATIC, DYNAMIC

Default value: DYNAMIC

submitMethod

HTTP method (GET or POST) used by the single sign-on form to send the logon credentials to the logon server.

Possible values: GET, POST

Default value: GET

unset vpn formSSOAction

Use this command to remove vpn formSSOAction settings. Refer to the set vpn formSSOAction command for meanings of the arguments.

Synopsis

unset vpn formSSOAction <name> [-responsesize] [-nameValuePair] [-nvtype] [-submitMethod]

show vpn formSSOAction

Displays the attributes of a form-based single sign-on profile.

Synopsis

show vpn formSSOAction [<name>]

Arguments

name

Name of the form-based single sign-on profile.

Outputs

actionURL

Root-relative URL to which the completed form is submitted.

userField

Username field.

passwdField

Password field.

responsesize

Maximum number of bytes to allow in the response size. Specifies the number of bytes in the response to be parsed for extracting the forms.

nameValuePair

Form attributes and their values to be submitted.

nvttype

Bypass Form extraction

ssoSuccessRule

Rule to be evaluated to check whether sso succeeded or not.

submitMethod

Form Submit method.

devno**count****stateflag**

vpn global

The following operations can be performed on "vpn global":

[bind](#) | [unbind](#) | [show](#)

bind vpn global

Binds NetScaler Gateway entities, including policies, globally.

Synopsys

```
bind vpn global [-policyName <string> [-priority <positive_integer>] [-secondary] [-groupExtraction]] [-intranetDomain <string>] [-intranetApplication <string>] [-nextHopServer <string>] [-urlName <string>] [-intranetIP <ip_addr> <netmask>] [-staServer <URL> [-staAddressType ( IPV4 | IPV6 )]] [-appController <URL>] [-sharefile <string>]
```

Arguments

policyName

Name of the policy to bind globally.

priority

Priority assigned to this session policy. A lower number indicates a higher priority.

Maximum value for default syntax policies is 4294967295 and for classic policies is 64000.

Minimum value: 0

secondary

Bind the authentication policy as the secondary policy to use in a two-factor configuration. A user must then authenticate not only to a primary authentication server but also to a secondary authentication server. User groups are aggregated across both authentication servers. The user name must be exactly the same on both authentication servers, but the authentication servers can require different passwords.

groupExtraction

Bind the Authentication policy to a tertiary chain which will be used only for group extraction. The user will not authenticate against this server, and this will only be called if primary and/or secondary authentication has succeeded.

intranetDomain

Intranet domain name for single sign-on.

intranetApplication

Name of the intranet application to bind globally.

nextHopServer

Name of the next hop server to bind globally.

urlName

Name of the URL of the virtual server to bind globally.

intranetIP

Range of IP addresses in an address pool or individual IP addresses to bind globally.

netmask

The intranet ip or range's netmask.

staServer

Web address of the Secure Ticketing Authority (STA) server to be bound globally, in the following format: 'http(s)://FQDN/URLPATH'

staAddressType

Type of the STA server address(ipv4/v6).

Possible values: IPV4, IPV6

appController

App Controller server, in the format 'http(s)://IP/FQDN'

sharefile

ShareFile server, in the format 'IP:PORT / FQDN:PORT'

unbind vpn global

Unbinds NetScaler Gateway policies to the virtual server globally.

Synopsys

unbind vpn global [-policyName <string> [-secondary] [-groupExtraction]] [-intranetDomain <string>] [-intranetApplication <string>] [-nextHopServer <string>] [-urlName <string>] [-intranetIP <ip_addr> <netmask>] [-staServer <URL>] [-appController <URL>] [-sharefile <string>]

Arguments

policyName

Name of the policy to unbind globally.

secondary

Bind the authentication policy as the secondary policy to use in a two-factor configuration. A user must then authenticate not only to a primary authentication server but also to a secondary authentication server. User groups are aggregated across both authentication servers. The user name must be exactly the same on both authentication servers, but the authentication servers can require different passwords.

groupExtraction

Bind the Authentication policy to a tertiary chain which will be used only for group extraction. The user will not authenticate against this server, and this will only be called if primary and/or secondary authentication has succeeded.

intranetDomain

A conflicting intranet domain name to be unbound.

intranetApplication

The name of a VPN intranet application to be unbound.

nextHopServer

The name of the next hop server to be unbound globally.

urlName

The name of a VPN url to be unbound from vpn global.

intranetIP

The intranet IP address or range to be unbound.

netmask

The intranet IP or range's netmask to be unbound from vpn global.

staServer

Secure Ticketing Authority (STA) server to be removed, in the format 'http(s)://IP/FQDN/URLPATH'

appController

App Controller server to be removed, in the format 'http(s)://IP/FQDN'

sharefile

ShareFile server to be removed, in the format 'IP:PORT / FQDN:PORT'

show vpn global

Shows the NetScaler Gateway policies that are bound to the virtual server globally.

Synopsys

show vpn global

Outputs

stateflag

policyName

The name of the policy.

priority

The priority of the policy.

intranetDomain

The conflicting intranet domain name.

intranetApplication

The intranet vpn application.

nextHopServer

The name of the next hop server bound to vpn global.

urlName

The intranet url.

intranetIP

The intranet ip address or range.

netmask

The intranet ip address or range's netmask.

staServer

Configured Secure Ticketing Authority (STA) server.

staAddressType

Type of the STA server address(ipv4/v6).

staAuthID

Authority ID of the STA Server. Authority ID is used to match incoming STA Tickets in the SOCKS/CGP protocol with the right STA Server.

appController

Configured App Controller server.

sharefile

Configured Sharefile server, in the format IP:PORT / FQDN:PORT

type

Bindpoint to which the policy is bound

policySubType

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

secondary

Bind the authentication policy as the secondary policy to use in a two-factor configuration. A user must then authenticate not only to a primary authentication server but also to a secondary authentication server. User groups are aggregated across both authentication servers. The user name must be exactly the same on both authentication servers, but the authentication servers can require different passwords.

groupExtraction

Bind the Authentication policy to a tertiary chain which will be used only for group extraction. The user will not authenticate against this server, and this will only be called if primary and/or secondary authentication has succeeded.

policyType

Policy type (Classic/Advanced) to be bound.Used for display.

devno

count

vpn icaConnection

The following operations can be performed on "vpn icaConnection":

show vpn icaConnection

Displays active connections that use the ICA proxy.

Synopsys

```
show vpn icaConnection [-userName <string>]
```

Arguments

userName

User name for which to display connections.

Outputs

domain

The domain name.

srcIP

The client IP address.

srcPort

The client port.

destIP

The CPS server IP address.

destPort

The CPS server port.

peld

Core id of the session owner

stateflag

devno

count

vpn intranetApplication

The following operations can be performed on "vpn intranetApplication":

[add](#) | [rm](#) | [show](#)

add vpn intranetApplication

Defines intranet applications to be made accessible through NetScaler Gateway.

Synopsis

```
add vpn intranetApplication <intranetApplication> [<protocol>] ((<destIP> [-netmask <netmask>]) | <IPRange> | <hostName>) [-destPort <port[-port]>] [-interception ( PROXY | TRANSPARENT ) [-srcIP <ip_addr>] [-srcPort <port>]]
```

Arguments

intranetApplication

Name of the intranet application.

protocol

Protocol used by the intranet application. If protocol is set to BOTH, TCP and UDP traffic is allowed.

Possible values: TCP, UDP, ANY

destIP

Destination IP address, IP range, or host name of the intranet application. This address is the server IP address.

netmask

Destination subnet mask for the intranet application.

Default value: 0xFFFFFFFF

IPRange

If you have multiple servers in your network, such as web, email, and file shares, configure an intranet application that includes the IP range for all the network applications. This allows users to access all the intranet applications contained in the IP address range.

hostName

Name of the host for which to configure interception. The names are resolved during interception when users log on with the NetScaler Gateway Plug-in.

destPort

Destination TCP or UDP port number for the intranet application. Use a hyphen to specify a range of port numbers, for example 90-95.

Minimum value: 1

interception

Interception mode for the intranet application or resource. Correct value depends on the type of client software used to make connections. If the interception mode is set to TRANSPARENT, users connect with the NetScaler Gateway Plug-in for Windows. With the PROXY setting, users connect with the NetScaler Gateway Plug-in for Java.

Possible values: PROXY, TRANSPARENT

srcIP

Source IP address. Required if interception mode is set to PROXY. Default is the loopback address, 127.0.0.1.

srcPort

Source port for the application for which the NetScaler Gateway virtual server proxies the traffic. If users are connecting from a device that uses the NetScaler Gateway Plug-in for Java, applications must be configured manually by using the source IP address and TCP port values specified in the intranet application profile. If a port value is not set, the destination port value is used.

Minimum value: 1

rm vpn intranetApplication

Removes a configured intranet resource.

Synopsys

```
rm vpn intranetApplication <intranetApplication>
```

Arguments

intranetApplication

Name of the intranet resource to remove.

show vpn intranetApplication

Displays information about all the configured intranet resources, or displays detailed information about the specified intranet resource.

Synopsys

```
show vpn intranetApplication [<intranetApplication>]
```

Arguments

intranetApplication

Name of the intranet resource for which to display detailed information.

Outputs

protocol

The IP protocol; for example, TCP, UDP or ANY

destIP

The destination IP address.

netmask

The destination netmask.

IPAddress

The IP address for the application. This address is the real application server IP address.

hostName

Name based interception. Names should be valid dns or wins names and will be resolved during interception on the sslvpn.

destPort

The destination port.

clientApplication

Names of the client applications, such as PuTTY and Xshell.

spoofIP

This specifies whether to spoof this application on the client.

interception

The interception type; for example, proxy or transparent.

srcIP

The source IP address.

srcPort

The source port.

stateflag**devno****count**

vpn nextHopServer

The following operations can be performed on "vpn nextHopServer":

[add](#) | [rm](#) | [show](#)

add vpn nextHopServer

Enables a NetScaler Gateway appliance in the first DMZ to communicate with one or more NetScaler Gateway appliances in the second DMZ.

Synopsys

```
add vpn nextHopServer <name> <nextHopIP> <nextHopPort> [-secure ( ON | OFF )]
```

Arguments

name

Name for the NetScaler Gateway appliance in the first DMZ.

Maximum value: 32

nextHopIP

IP address or FQDN of the NetScaler Gateway proxy in the second DMZ.

nextHopPort

Port number of the NetScaler Gateway proxy in the second DMZ.

Minimum value: 1

Maximum value: 65535

secure

Use of a secure port, such as 443, for the double-hop configuration.

Possible values: ON, OFF

Default value: OFF

Example

```
add vpn nexthopserver dh1 10.1.1.1 80 -secure OFF
```

rm vpn nextHopServer

Removes a configured next hop server.

Synopsys

```
rm vpn nextHopServer <name>
```

Arguments

name

Name of the next hop server to remove.

Maximum value: 32

Example

```
rm vpn nexthopserver dh1
```

show vpn nextHopServer

Displays information about all the configured next NetScaler Gateway hop servers, or detailed information about the specified NetScaler Gateway next hop server.

Synopsys

```
show vpn nextHopServer [<name>]
```

Arguments

name

Name of the NetScaler Gateway next hop server for which to display detailed information.

Maximum value: 32

Outputs

nextHopIP

Next hop IP address.

nextHopPort

Next hop port number.

secure

Next hop over secure connection.

stateflag

devno

count

Example

```
show vpn nexthopserver dh1
```

vpn parameter

The following operations can be performed on "vpn parameter":

[set](#) | [unset](#) | [show](#)

set vpn parameter

Sets global parameters for NetScaler Gateway.

Synopsys

```
set vpn parameter [-httpPort <port> ...] [-winsIP <ip_addr>] [-dnsVserverName <string>] [-splitDns <splitDns>] [-sessTimeout <mins>] [-clientSecurity <expression>] [-clientSecurityGroup <string>] [-clientSecurityMessage <string>] [-clientSecurityLog ( ON | OFF )] [-splitTunnel <splitTunnel>] [-localLanAccess ( ON | OFF )] [-rfc1918 ( ON | OFF )] [-killConnections ( ON | OFF )] [-transparentInterception ( ON | OFF )] [-defaultAuthorizationAction ( ALLOW | DENY )] [-authorizationGroup <string>] [-clientIdleTimeout <mins>] [-proxy <proxy>] [-allProtocolProxy <string>] [-httpProxy <string>] [-ftpProxy <string>] [-socksProxy <string>] [-gopherProxy <string>] [-sslProxy <string>] [-proxyException <string>] [-proxyLocalBypass ( ENABLED | DISABLED )] [-clientCleanupPrompt ( ON | OFF )] [-forceCleanup <forceCleanup> ...] [-clientOptions <clientOptions> ...] [-clientConfiguration <clientConfiguration> ...] [-SSO ( ON | OFF )] [-ssoCredential ( PRIMARY | SECONDARY )] [-windowsAutoLogon ( ON | OFF )] [-useMIP ( NS | OFF )] [-useIIP <useIIP>] [-clientDebug <clientDebug>] [-loginScript <input_filename>] [-logoutScript <input_filename>] [-homePage <URL>] [-icaProxy ( ON | OFF )] [-wihome <URL>] [-wihomeAddressType ( IPV4 | IPV6 )] [-citrixReceiverHome <URL>] [-wiPortalMode ( NORMAL | COMPACT )] [-ClientChoices ( ON | OFF )] [-iipDnsSuffix <string>] [-forcedTimeout <mins>] [-forcedTimeoutWarning <mins>] [-ntDomain <string>] [-clientlessVpnMode <clientlessVpnMode>] [-clientlessModeUrlEncoding <clientlessModeUrlEncoding>] [-clientlessPersistentCookie <clientlessPersistentCookie>] [-emailHome <URL>] [-allowedLoginGroups <string>] [-encryptCsecExp ( ENABLED | DISABLED )] [-appTokenTimeout <positive_integer>] [-mdxTokenTimeout <positive_integer>] [-UI THEME <UI THEME>] [-SecureBrowse ( ENABLED | DISABLED )] [-storefronturl <string>] [-kcdAccount <string>]
```

Arguments

httpPort

Destination port numbers other than port 80, added as a comma-separated list. Traffic to these ports is processed as HTTP traffic, which allows functionality, such as HTTP authorization and single sign-on to a web application to work.

Minimum value: 1

winsIP

WINS server IP address to add to NetScaler Gateway for name resolution.

dnsVserverName

Name of the DNS virtual server for the user session.

splitDns

Route the DNS requests to the local DNS server configured on the user device, or NetScaler Gateway (remote), or both.

Possible values: LOCAL, REMOTE, BOTH

sessTimeout

Number of minutes after which the session times out.

Default value: 30

Minimum value: 1

Maximum value: 65535

clientSecurity

Specify the client security check for the user device to permit a NetScaler Gateway session. The web address or IP address is not included in the expression for the client security check.

clientSecurityGroup

The client security group that will be assigned on failure of the client security check. Users can in general be organized into Groups. In this case, the Client Security Group may have a more restrictive security policy.

clientSecurityMessage

The client security message that will be displayed on failure of the client security check.

clientSecurityLog

Set the logging of client security checks.

Possible values: ON, OFF

Default value: ON

splitTunnel

Send, through the tunnel, traffic only for intranet applications that are defined in NetScaler Gateway. Route all other traffic directly to the Internet. The OFF setting routes all traffic through NetScaler Gateway. With the REVERSE setting, intranet applications define the network traffic that is not intercepted. All network traffic directed to internal IP addresses bypasses the VPN tunnel, while other traffic goes through NetScaler Gateway. Reverse split tunneling can be used to log all non-local LAN traffic. For example, if users have a home network and are logged on through the NetScaler Gateway Plug-in, network traffic destined to a printer or another device within the home network is not intercepted.

Possible values: ON, OFF, REVERSE

Default value: OFF

localLanAccess

Set local LAN access. If split tunneling is OFF, and you set local LAN access to ON, the local client can route traffic to its local interface. When the local area network switch is specified, this combination of switches is useful. The client can allow local LAN access to devices that commonly have non-routable addresses, such as local printers or local file servers.

Possible values: ON, OFF

Default value: OFF

rfc1918

As defined in the local area network, allow only the following local area network addresses to bypass the VPN tunnel when the local LAN access feature is enabled:

* 10.*.* ,

* 172.16.*.* ,

* 192.168.*.*

Possible values: ON, OFF

Default value: OFF

killConnections

Specify whether the NetScaler Gateway Plug-in should disconnect all preexisting connections, such as the connections existing before the user logged on to NetScaler Gateway, and prevent new incoming connections on the NetScaler Gateway Plug-in for Windows and MAC when the user is connected to NetScaler Gateway and split tunneling is disabled.

Possible values: ON, OFF

Default value: OFF

transparentInterception

Allow access to network resources by using a single IP address and subnet mask or a range of IP addresses. The OFF setting sets the mode to proxy, in which you configure destination and source IP addresses and port numbers. If you are using the NetScaler Gateway Plug-in for Windows, set this parameter to ON, in which the mode is set to transparent. If you are using the NetScaler Gateway Plug-in for Java, set this parameter to OFF.

Possible values: ON, OFF

Default value: ON

defaultAuthorizationAction

Specify the network resources that users have access to when they log on to the internal network. The default setting for authorization is to deny access to all network resources. Citrix recommends using the default global setting and then creating authorization policies to define the network resources users can access. If you set the default authorization policy to DENY, you must explicitly authorize access to any network resource, which improves security.

Possible values: ALLOW, DENY

Default value: DENY

authorizationGroup

Comma-separated list of groups in which the user is placed when none of the groups that the user is a part of is configured on NetScaler Gateway. The authorization policy can be bound to these groups to control access to the resources.

clientIdleTimeout

Time, in minutes, after which to time out the user session if NetScaler Gateway does not detect mouse or keyboard activity.

Minimum value: 1

Maximum value: 9999

proxy

Set options to apply proxy for accessing the internal resources. Available settings function as follows:

* BROWSER - Proxy settings are configured only in Internet Explorer and Firefox browsers.

* NS - Proxy settings are configured on the NetScaler appliance.

* OFF - Proxy settings are not configured.

Possible values: BROWSER, NS, OFF

allProtocolProxy

IP address of the proxy server to use for all protocols supported by NetScaler Gateway.

httpProxy

IP address of the proxy server to be used for HTTP access for all subsequent connections to the internal network.

ftpProxy

IP address of the proxy server to be used for FTP access for all subsequent connections to the internal network.

socksProxy

IP address of the proxy server to be used for SOCKS access for all subsequent connections to the internal network.

gopherProxy

IP address of the proxy server to be used for GOPHER access for all subsequent connections to the internal network.

sslProxy

IP address of the proxy server to be used for SSL access for all subsequent connections to the internal network.

proxyException

Proxy exception string that will be configured in the browser for bypassing the previously configured proxies. Allowed only if proxy type is Browser.

proxyLocalBypass

Bypass proxy server for local addresses option in Internet Explorer and Firefox proxy server settings.

Possible values: ENABLED, DISABLED

Default value: DISABLED

clientCleanupPrompt

Prompt for client-side cache clean-up when a client-initiated session closes.

Possible values: ON, OFF

Default value: ON

forceCleanup

Force cache clean-up when the user closes a session. You can specify all, none, or any combination of the client-side items.

clientOptions

Display only the configured menu options when you select the "Configure NetScaler Gateway" option in the NetScaler Gateway Plug-in's system tray icon for Windows.

clientConfiguration

Display only the configured tabs when you select the "Configure NetScaler Gateway" option in the NetScaler Gateway Plug-in's system tray icon for Windows.

SSO

Set single sign-on (SSO) for the session. When the user accesses a server, the user's logon credentials are passed to the server for authentication.

Possible values: ON, OFF

Default value: OFF

ssoCredential

Specify whether to use the primary or secondary authentication credentials for single sign-on to the server.

Possible values: PRIMARY, SECONDARY

Default value: PRIMARY

windowsAutoLogon

Enable or disable the Windows Auto Logon for the session. If a VPN session is established after this setting is enabled, the user is automatically logged on by using Windows credentials after the system is restarted.

Possible values: ON, OFF

Default value: OFF

useMIP

Enable or disable the use of a unique IP address alias, or a mapped IP address, as the client IP address for each client session. Allow NetScaler Gateway to use the mapped IP address as an intranet IP address when all other IP addresses are not available.

When IP pooling is configured and the mapped IP is used as an intranet IP address, the mapped IP address is used when an intranet IP address cannot be assigned.

Possible values: NS, OFF

Default value: NS

useIIP

Define IP address pool options. Available settings function as follows:

* SPILLOVER - When an address pool is configured and the mapped IP is used as an intranet IP address, the mapped IP address is used when an intranet IP address cannot be assigned.

* NOSPILOVER - When intranet IP addresses are enabled and the mapped IP address is not used, the Transfer Login page appears for users who have used all available intranet IP addresses.

* OFF - Address pool is not configured.

Possible values: NOSPILOVER, SPILLOVER, OFF

Default value: NOSPILOVER

clientDebug

Set the trace level on NetScaler Gateway. Technical support technicians use these debug logs for in-depth debugging and troubleshooting purposes. Available settings function as follows:

* DEBUG - Detailed debug messages are collected and written into the specified file.

* STATS - Application audit level error messages and debug statistic counters are written into the specified file.

* EVENTS - Application audit-level error messages are written into the specified file.

* OFF - Only critical events are logged into the Windows Application Log.

Possible values: debug, stats, events, OFF

Default value: OFF

loginScript

Path to the logon script that is run when a session is established. Separate multiple scripts by using comma. A "\$" in the path signifies that the word following the "\$" is an environment variable.

logoutScript

Path to the logout script. Separate multiple scripts by using comma. A "\$" in the path signifies that the word following the "\$" is an environment variable.

homePage

Web address of the home page that appears when users log on. Otherwise, users receive the default home page for NetScaler Gateway, which is the Access Interface.

icaProxy

Enable ICA proxy to configure secure Internet access to servers running Citrix XenApp or XenDesktop by using Citrix Receiver instead of the NetScaler Gateway Plug-in.

Possible values: ON, OFF

Default value: OFF

wihome

Web address of the Web Interface server, such as `http://<ipAddress>/Citrix/XenApp`, or Receiver for Web, which enumerates the virtualized resources, such as XenApp, XenDesktop, and cloud applications. This web address is used as the home page in ICA proxy mode.

If Client Choices is ON, you must configure this setting. Because the user can choose between FullClient and ICAProxy, the user may see a different home page. An Internet web site may appear if the user gets the FullClient option, or a Web Interface site if the user gets the ICAProxy option. If the setting is not configured, the XenApp option does not appear as a client choice.

wihomeAddressType

Type of the wihome address(IPV4/V6)

Possible values: IPV4, IPV6

citrixReceiverHome

Web address for the Citrix Receiver home page. Configure NetScaler Gateway so that when users log on to the appliance, the NetScaler Gateway Plug-in opens a web browser that allows single sign-on to the Citrix Receiver home page.

wiPortalMode

Layout on the Access Interface. The COMPACT value indicates the use of small icons.

Possible values: NORMAL, COMPACT

ClientChoices

Provide users with multiple logon options. With client choices, users have the option of logging on by using the NetScaler Gateway Plug-in for Windows, NetScaler Gateway Plug-in for Java, the Web Interface, or clientless access from one location. Depending on how NetScaler Gateway is configured, users are presented with up to three icons for logon choices. The most common are the NetScaler Gateway Plug-in for Windows, Web Interface, and clientless access.

Possible values: ON, OFF

Default value: OFF

iipDnsSuffix

An intranet IP DNS suffix. When a user logs on to NetScaler Gateway and is assigned an IP address, a DNS record for the user name and IP address combination is added to the NetScaler Gateway DNS cache. You can configure a DNS suffix to append to the user name when the DNS record is added to the cache. You can reach to the host from where the user is logged on by using the user's name, which can be easier to remember than an IP address. When the user logs off from NetScaler Gateway, the record is removed from the DNS cache.

forcedTimeout

Force a disconnection from the NetScaler Gateway Plug-in with NetScaler Gateway after a specified number of minutes. If the session closes, the user must log on again.

Minimum value: 1

Maximum value: 65535

forcedTimeoutWarning

Number of minutes to warn a user before the user session is disconnected.

Minimum value: 1

Maximum value: 255

ntDomain

Single sign-on domain to use for single sign-on to applications in the internal network. This setting can be overwritten by the domain that users specify at the time of logon or by the domain that the authentication server returns.

clientlessVpnMode

Enable clientless access for web, XenApp or XenDesktop, and FileShare resources without installing the NetScaler Gateway Plug-in. Available settings function as follows:

- * ON - Allow only clientless access.
- * OFF - Allow clientless access after users log on with the NetScaler Gateway Plug-in.
- * DISABLED - Do not allow clientless access.

Possible values: ON, OFF, DISABLED

Default value: OFF

clientlessModeUrlEncoding

When clientless access is enabled, you can choose to encode the addresses of internal web applications or to leave the address as clear text. Available settings function as follows:

- * OPAQUE - Use standard encoding mechanisms to make the domain and protocol part of the resource unclear to users.
- * TRANSPARENT - Do not encode the web address and make it visible to users.
- * ENCRYPT - Allow the domain and protocol to be encrypted using a session key. When the web address is encrypted, the URL is different for each user session for the same web resource. If users bookmark the encoded web address, save it in the web browser and then log off, they cannot connect to the web address when they log on and use the bookmark. If users save the encrypted bookmark in the Access Interface during their session, the bookmark works each time the user logs on.

Possible values: TRANSPARENT, OPAQUE, ENCRYPT

Default value: OPAQUE

clientlessPersistentCookie

State of persistent cookies in clientless access mode. Persistent cookies are required for accessing certain features of SharePoint, such as opening and editing Microsoft Word, Excel, and PowerPoint documents hosted on the SharePoint server. A persistent cookie remains on the user device and is sent with each HTTP request. NetScaler Gateway encrypts the persistent cookie before sending it to the plug-in on the user device, and refreshes the cookie periodically as long as the session exists. The cookie becomes stale if the session ends. Available settings function as follows:

- * ALLOW - Enable persistent cookies. Users can open and edit Microsoft documents stored in SharePoint.
- * DENY - Disable persistent cookies. Users cannot open and edit Microsoft documents stored in SharePoint.
- * PROMPT - Prompt users to allow or deny persistent cookies during the session. Persistent cookies are not required for clientless access if users do not connect to SharePoint.

Possible values: ALLOW, DENY, PROMPT

Default value: DENY

emailHome

Web address for the web-based email, such as Outlook Web Access.

allowedLoginGroups

Specify groups that have permission to log on to NetScaler Gateway. Users who do not belong to this group or groups are denied access even if they have valid credentials.

encryptCsecExp

Enable encryption of client security expressions.

Possible values: ENABLED, DISABLED

Default value: ENABLED

appTokenTimeout

The timeout value in seconds for tokens to access XenMobile applications

Default value: 100

Minimum value: 1

Maximum value: 255

mdxTokenTimeout

Validity of MDX Token in minutes. This token is used for mdx services to access backend and valid HEAD and GET request.

Default value: 10

Minimum value: 1

Maximum value: 1440

UITHEME

Set VPN UI Theme to Green-Bubble, Caxton or Custom; default is Caxton.

Possible values: DEFAULT, GREENBUBBLE, CUSTOM

SecureBrowse

Allow users to connect through NetScaler Gateway to network resources from iOS and Android mobile devices with Citrix Receiver. Users do not need to establish a full VPN tunnel to access resources in the secure network.

Possible values: ENABLED, DISABLED

Default value: ENABLED

storefronturl

Web address for StoreFront to be used in this session for enumeration of resources from XenApp or XenDesktop.

kcdAccount

The KCD account details to be used in SSO

Example

```
set vpn parameter -httpport 80 90 -winsIP 192.168.0.220 -dnsVserverName mydns -sessTimeout
```

unset vpn parameter

Removes global parameters for NetScaler Gateway..Refer to the set vpn parameter command for meanings of the arguments.

Synopsys

```
unset vpn parameter [-httpPort] [-winsIP] [-dnsVserverName] [-splitDns] [-sessTimeout] [-clientSecurity] [-clientSecurityGroup] [-clientSecurityMessage] [-clientSecurityLog] [-authorizationGroup] [-clientIdleTimeout] [-allProtocolProxy] [-httpProxy] [-ftpProxy] [-socksProxy] [-gopherProxy] [-sslProxy] [-proxyException] [-forceCleanup] [-clientOptions] [-clientConfiguration] [-loginScript] [-logoutScript] [-homePage] [-proxy] [-wihome] [-citrixReceiverHome] [-wiPortalMode] [-iipDnsSuffix] [-forcedTimeout] [-forcedTimeoutWarning] [-defaultAuthorizationAction] [-ntDomain] [-clientlessVpnMode] [-emailHome] [-clientlessModeUrlEncoding] [-clientlessPersistentCookie] [-allowedLoginGroups] [-appTokenTimeout] [-mdxTokenTimeout] [-storefronturl] [-UITHEME] [-kcdAccount] [-splitTunnel] [-localLanAccess] [-rfc1918] [-killConnections] [-transparentInterception] [-proxyLocalBypass] [-clientCleanupPrompt] [-SSO] [-ssoCredential] [-windowsAutoLogon] [-useMIP] [-useIIP] [-clientDebug] [-icaProxy] [-ClientChoices] [-encryptCsecExp] [-SecureBrowse]
```

show vpn parameter

Displays the configured NetScaler Gateway parameters.

Synopsys

show vpn parameter

Outputs

name

The VPN name.

httpPort

The HTTP Port.

winsIP

The WINS server IP address used for WINS host resolution by the VPN.

dnsVserverName

The configured DNS vserver used for DNS host resolution by the VPN.

splitDns

The VPN client SplitDns state.

sessTimeout

The session timeout, in minutes.

clientSecurity

The client security check applied to client sessions. This is in the form of an expression. Expressions are simple conditions, such as a test for equality, applied to operands, such as a URL string or an IP address. Expression syntax is described in the Installation and Configuration Guide.

clientSecurityGroup

The client security group that will be assigned on failure of the client security check. Users can in general be organized into Groups. In this case, the Client Security Group may have a more restrictive security policy.

clientSecurityMessage

The client security message that will be displayed on failure of the client security check.

clientSecurityLog

Set the logging of client security checks.

splitTunnel

Send, through the tunnel, traffic only for intranet applications that are defined in NetScaler Gateway. Route all other traffic directly to the Internet. The OFF setting routes all traffic through NetScaler Gateway. With the REVERSE setting, intranet applications define the network traffic that is not intercepted. All network traffic directed to internal IP addresses bypasses the VPN tunnel, while other traffic goes through NetScaler Gateway. Reverse split tunneling can be used to log all non-local LAN traffic. For example, if users have a home network and are logged on through the NetScaler Gateway Plug-in, network traffic destined to a printer or another device within the home network is not intercepted.

localLanAccess

Set local LAN access. If split tunneling is OFF, and you set local LAN access to ON, the local client can route traffic to its local interface. When the local area network switch is specified, this combination of switches is useful. The client can allow local LAN access to devices that commonly have non-routable addresses, such as local printers or local file servers.

rfc1918

Only allow RFC1918 local addresses when local LAN access feature is enabled.

spoofIP

Controls the Spoofing of Intranet IP to the Windows Applications by Windows VPN client when the end-user is connected to SSL VPN in '-splittunnel OFF' mode.

killConnections

Determines whether Windows VPN client should kill all pre-existing connections; (for example, the connections existing before the end user logged in to SSL VPN) and prevent new incoming connections on the Windows Client system when the end user is connected to SSL VPN in '-splittunnel OFF' mode.

transparentInterception

Allow access to network resources by using a single IP address and subnet mask or a range of IP addresses. The OFF setting sets the mode to proxy, in which you configure destination and source IP addresses and port numbers. If you are using the NetScaler Gateway Plug-in for Windows, set this parameter to ON, in which the mode is set to transparent. If you are using the NetScaler Gateway Plug-in for Java, set this parameter to OFF.

windowsClientType

The windows client type.

defaultAuthorizationAction

The Authentication Action, such as allow or deny.

authorizationGroup

The authorization group applied to the session.

clientIdleTimeout

The client idle timeout, in minutes.

clientIdleTimeoutWarning

The time after which the client gets a timeout warning, in minutes.

proxy

Proxy configuration for the session.

allProtocolProxy

Address set for all proxies.

httpProxy

IP address of the proxy server to be used for HTTP access for all subsequent connections to the internal network.

ftpProxy

IP address of the proxy server to be used for FTP access for all subsequent connections to the internal network.

socksProxy

IP address of the proxy server to be used for SOCKS access for all subsequent connections to the internal network.

gopherProxy

IP address of the proxy server to be used for GOPHER access for all subsequent connections to the internal network.

sslProxy

IP address of the proxy server to be used for SSL access for all subsequent connections to the internal network.

proxyException

The Proxy Exception string that is configured in the Browser for bypassing the previously configured proxies. Allowed only if proxy type is Browser.

proxyLocalBypass

Bypass proxy server for local addresses option in Internet Explorer and Firefox proxy server settings.

clientCleanupPrompt

Prompt for client-side cache clean-up when a client-initiated session closes.

forceCleanup

Whether or not to force a cleanup on exit from the VPN session.

clientOptions

List of configured buttons(and/or menu options in the docked client) in the Windows VPN client.

clientConfiguration

List of configured tabs in the Windows VPN client.

SSO

Enable or Disable Single Sign-On.

ssoCredential

Specify whether to use the primary or secondary authentication credentials for single sign-on to the server.

windowsAutoLogon

Enable or Disable Windows Auto Logon.

useMIP

Enables or disables the use of a Mapped IP address for the session.

useIP

Define IP address pool options. Available settings function as follows:

* SPILLOVER - When an address pool is configured and the mapped IP is used as an intranet IP address, the mapped IP address is used when an intranet IP address cannot be assigned.

* NOSPILOVER - When intranet IP addresses are enabled and the mapped IP address is not used, the Transfer Login page appears for users who have used all available intranet IP addresses.

* OFF - Address pool is not configured.

clientDebug

Whether or not to add debugging information to the activity log on the client.

loginScript

Path to the logon script that is run when a session is established. Separate multiple scripts by using comma. A "\$" in the path signifies that the word following the "\$" is an environment variable.

logoutScript

Path to the logout script. Separate multiple scripts by using comma. A "\$" in the path signifies that the word following the "\$" is an environment variable.

homePage

The home page URL, or 'none'. 'none' is case sensitive.

icaProxy

Enable ICA proxy to configure secure Internet access to servers running Citrix XenApp or XenDesktop by using Citrix Receiver instead of the NetScaler Gateway Plug-in.

wihome

Web address of the Web Interface server, such as `http://<ipAddress>/Citrix/XenApp`, or Receiver for Web, which enumerates the virtualized resources, such as XenApp, XenDesktop, and cloud applications. This web address is used as the home page in ICA proxy mode.

If Client Choices is ON, you must configure this setting. Because the user can choose between FullClient and ICAProxy, the user may see a different home page. An Internet web site may appear if the user gets the FullClient option, or a Web Interface site if the user gets the ICAProxy option. If the setting is not configured, the XenApp option does not appear as a client choice.

wihomeAddressType

Type of the wihome address(IPV4/V6)

citrixReceiverHome

Web address for the Citrix Receiver home page. Configure NetScaler Gateway so that when users log on to the appliance, the NetScaler Gateway Plug-in opens a web browser that allows single sign-on to the Citrix Receiver home page.

wiPortalMode

Layout on the Access Interface. The COMPACT value indicates the use of small icons.

ClientChoices

Provide users with multiple logon options. With client choices, users have the option of logging on by using the NetScaler Gateway Plug-in for Windows, NetScaler Gateway Plug-in for Java, the Web Interface, or clientless access from one location. Depending on how NetScaler Gateway is configured, users are presented with up to three icons for logon choices. The most common are the NetScaler Gateway Plug-in for Windows, Web Interface, and clientless access.

epaClientType

Choose between two types of End point Windows Client

a) Application Agent - which always runs in the task bar as a standalone application and also has a supporting service which runs permanently when installed

b) Activex Control - ActiveX control run by Microsoft Internet Explorer.

iipDnsSuffix

The DNS suffix for the intranet IP address.

forcedTimeout

The time, in minutes after which a timeout is forced.

forcedTimeoutWarning

The time, in minutes, after which a timeout warning is issued.

ntDomain

Single sign-on domain to use for single sign-on to applications in the internal network. This setting can be overwritten by the domain that users specify at the time of logon or by the domain that the authentication server returns.

clientlessVpnMode

Whether clientless VPN is available to the session.

clientlessModeUrlEncoding

URL encoding to be used for clientless mode.

clientlessPersistentCookie

State of persistent cookies in clientless access mode. Persistent cookies are required for accessing certain features of SharePoint, such as opening and editing Microsoft Word, Excel, and PowerPoint documents hosted on the SharePoint server. A persistent cookie remains on the user device and is sent with each HTTP request. NetScaler Gateway encrypts the persistent cookie before sending it to the plug-in on the user device, and refreshes the cookie periodically as long as the session exists. The cookie becomes stale if the session ends. Available settings function as follows:

- * ALLOW - Enable persistent cookies. Users can open and edit Microsoft documents stored in SharePoint.
- * DENY - Disable persistent cookies. Users cannot open and edit Microsoft documents stored in SharePoint.
- * PROMPT - Prompt users to allow or deny persistent cookies during the session. Persistent cookies are not required for clientless access if users do not connect to SharePoint.

emailHome

Web address for the web-based email, such as Outlook Web Access.

allowedLoginGroups

Specify groups that have permission to log on to NetScaler Gateway. Users who do not belong to this group or groups are denied access even if they have valid credentials.

encryptCsecExp

Enable encryption of client security expressions.

appTokenTimeout

The timeout value in seconds for tokens to access XenMobile applications

mdxTokenTimeout

Validity of MDX Token in minutes. This token is used for mdx services to access backend and valid HEAD and GET request.

UITHEME

Set VPN UI Theme to Green-Bubble, Caxton or Custom; default is Caxton.

SecureBrowse

Allow users to connect through NetScaler Gateway to network resources from iOS and Android mobile devices with Citrix Receiver. Users do not need to establish a full VPN tunnel to access resources in the secure network.

storefronturl

Web address for StoreFront to be used in this session for enumeration of resources from XenApp or XenDesktop.

kcdAccount

The KCD account details to be used in SSO

tag

vpn samlSSOProfile

The following operations can be performed on "vpn samlSSOProfile":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add vpn samlSSOProfile

Creates a SAML single sign-on profile. This profile is employed in triggering saml assertion to a target service based on traffic profile.

Synopsys

```
add vpn samlSSOProfile <name> [-samlSigningCertName <string>] -assertionConsumerServiceURL <URL> -
relaystateRule <expression> [-sendPassword ( ON | OFF )] [-samlIssuerName <string>] [-signatureAlg ( RSA-SHA1
| RSA-SHA256 )] [-digestMethod ( SHA1 | SHA256 )] [-audience <string>] [-NameIDFormat <NameIDFormat>] [-
NameIDExpr <string>] [-Attribute1 <string> -Attribute1Expr <string> [-Attribute1FriendlyName <string>] [-
Attribute1Format ( URI | Basic )]] [-Attribute2 <string> -Attribute2Expr <string> [-Attribute2FriendlyName <string>] [-
Attribute2Format ( URI | Basic )]] [-Attribute3 <string> -Attribute3Expr <string> [-Attribute3FriendlyName <string>] [-
Attribute3Format ( URI | Basic )]] [-Attribute4 <string> -Attribute4Expr <string> [-Attribute4FriendlyName <string>] [-
Attribute4Format ( URI | Basic )]] [-Attribute5 <string> -Attribute5Expr <string> [-Attribute5FriendlyName <string>] [-
Attribute5Format ( URI | Basic )]] [-Attribute6 <string> -Attribute6Expr <string> [-Attribute6FriendlyName <string>] [-
Attribute6Format ( URI | Basic )]] [-Attribute7 <string> -Attribute7Expr <string> [-Attribute7FriendlyName <string>] [-
Attribute7Format ( URI | Basic )]] [-Attribute8 <string> -Attribute8Expr <string> [-Attribute8FriendlyName <string>] [-
Attribute8Format ( URI | Basic )]] [-Attribute9 <string> -Attribute9Expr <string> [-Attribute9FriendlyName <string>] [-
Attribute9Format ( URI | Basic )]] [-Attribute10 <string> -Attribute10Expr <string> [-Attribute10FriendlyName
<string>] [-Attribute10Format ( URI | Basic )]] [-Attribute11 <string> -Attribute11Expr <string> [-
Attribute11FriendlyName <string>] [-Attribute11Format ( URI | Basic )]] [-Attribute12 <string> -Attribute12Expr
<string> [-Attribute12FriendlyName <string>] [-Attribute12Format ( URI | Basic )]] [-Attribute13 <string> -
Attribute13Expr <string> [-Attribute13FriendlyName <string>] [-Attribute13Format ( URI | Basic )]] [-Attribute14
<string> -Attribute14Expr <string> [-Attribute14FriendlyName <string>] [-Attribute14Format ( URI | Basic )]] [-
Attribute15 <string> -Attribute15Expr <string> [-Attribute15FriendlyName <string>] [-Attribute15Format ( URI | Basic
)]] [-Attribute16 <string> -Attribute16Expr <string> [-Attribute16FriendlyName <string>] [-Attribute16Format ( URI |
Basic )]]
```

Arguments

name

Name for the new saml single sign-on profile. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after an SSO action is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my action" or 'my action').

samlSigningCertName

Name of the signing authority as given in the SAML server's SSL certificate.

assertionConsumerServiceURL

URL to which the assertion is to be sent.

relaystateRule

Expression to extract relaystate to be sent along with assertion. Evaluation of this expression should return TEXT content. This is typically a target url to which user is redirected after the recipient validates SAML token

sendPassword

Option to send password in assertion.

Possible values: ON, OFF

Default value: OFF

samlIssuerName

The name to be used in requests sent from Netscaler to IdP to uniquely identify Netscaler.

signatureAlg

Algorithm to be used to sign/verify SAML transactions

Possible values: RSA-SHA1, RSA-SHA256

Default value: RSA-SHA1

digestMethod

Algorithm to be used to compute/verify digest for SAML transactions

Possible values: SHA1, SHA256

Default value: SHA1

audience

Audience for which assertion sent by IdP is applicable. This is typically entity name or url that represents ServiceProvider

Maximum value: 256

NameIDFormat

Format of Name Identifier sent in Assertion.

Possible values: Unspecified, emailAddress, X509SubjectName, WindowsDomainQualifiedName, kerberos, entity, persistent, transient

Default value: transient

NameIDExpr

Expression that will be evaluated to obtain NameIdentifier to be sent in assertion

Attribute1

Name of attribute1 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute1Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute1FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute1Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute2

Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute2Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute2FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute2Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute3

Name of attribute3 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute3Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute3FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute3Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute4

Name of attribute4 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute4Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute4FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute4Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute5

Name of attribute5 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute5Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute5FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute5Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute6

Name of attribute6 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute6Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute6FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute6Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute7

Name of attribute7 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute7Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute7FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute7Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute8

Name of attribute8 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute8Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute8FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute8Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute9

Name of attribute9 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute9Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute9FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute9Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute10

Name of attribute10 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute10Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute10FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute10Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute11

Name of attribute11 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute11Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute11FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute11Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute12

Name of attribute12 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute12Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute12FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute12Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute13

Name of attribute13 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute13Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute13FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute13Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute14

Name of attribute14 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute14Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute14FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute14Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute15

Name of attribute15 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute15Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute15FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute15Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute16

Name of attribute16 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute16Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute16FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute16Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

rm vpn samlSSOProfile

Deletes an existing saml single sign-on traffic profile.

Synopsys

rm vpn samlSSOProfile <name>

Arguments

name

Name for the new saml single sign-on profile. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after an SSO action is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my action" or 'my action').

set vpn samlSSOProfile

Modifies the specified attributes of a saml single sign-on traffic profile.

Synopsys

```
set vpn samlSSOProfile <name> [-samlSigningCertName <string>] [-assertionConsumerServiceURL <URL>] [-sendPassword ( ON | OFF )] [-samlIssuerName <string>] [-relaystateRule <expression>] [-signatureAlg ( RSA-SHA1 | RSA-SHA256 )] [-digestMethod ( SHA1 | SHA256 )] [-audience <string>] [-NameIDFormat <NameIDFormat>] [-NameIDExpr <string>] [-Attribute1 <string> -Attribute1Expr <string> [-Attribute1FriendlyName <string>] [-Attribute1Format ( URI | Basic )]] [-Attribute2 <string> -Attribute2Expr <string> [-Attribute2FriendlyName <string>] [-Attribute2Format ( URI | Basic )]] [-Attribute3 <string> -Attribute3Expr <string> [-Attribute3FriendlyName <string>] [-Attribute3Format ( URI | Basic )]] [-Attribute4 <string> -Attribute4Expr <string> [-Attribute4FriendlyName <string>] [-Attribute4Format ( URI | Basic )]] [-Attribute5 <string> -Attribute5Expr <string> [-Attribute5FriendlyName <string>] [-Attribute5Format ( URI | Basic )]] [-Attribute6 <string> -Attribute6Expr <string> [-Attribute6FriendlyName <string>] [-Attribute6Format ( URI | Basic )]] [-Attribute7 <string> -Attribute7Expr <string> [-Attribute7FriendlyName <string>] [-Attribute7Format ( URI | Basic )]] [-Attribute8 <string> -Attribute8Expr <string> [-Attribute8FriendlyName <string>] [-Attribute8Format ( URI | Basic )]] [-Attribute9 <string> -Attribute9Expr <string> [-Attribute9FriendlyName <string>] [-Attribute9Format ( URI | Basic )]] [-Attribute10 <string> -Attribute10Expr <string> [-Attribute10FriendlyName <string>] [-Attribute10Format ( URI | Basic )]] [-Attribute11 <string> -Attribute11Expr <string> [-Attribute11FriendlyName <string>] [-Attribute11Format ( URI | Basic )]] [-Attribute12 <string> -Attribute12Expr <string> [-Attribute12FriendlyName <string>] [-Attribute12Format ( URI | Basic )]] [-Attribute13 <string> -Attribute13Expr <string> [-Attribute13FriendlyName <string>] [-Attribute13Format ( URI | Basic )]] [-Attribute14 <string> -Attribute14Expr <string> [-Attribute14FriendlyName <string>] [-Attribute14Format ( URI | Basic )]] [-Attribute15 <string> -Attribute15Expr <string> [-Attribute15FriendlyName <string>] [-Attribute15Format ( URI | Basic )]] [-Attribute16 <string> -Attribute16Expr <string> [-Attribute16FriendlyName <string>] [-Attribute16Format ( URI | Basic )]]
```

Arguments

name

Name for the new saml single sign-on profile. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after an SSO action is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my action" or 'my action').

samlSigningCertName

Name of the signing authority as given in the SAML server's SSL certificate.

assertionConsumerServiceURL

URL to which the assertion is to be sent.

sendPassword

Option to send password in assertion.

Possible values: ON, OFF

Default value: OFF

samlIssuerName

The name to be used in requests sent from Netscaler to IdP to uniquely identify Netscaler.

relaystateRule

Expression to extract relaystate to be sent along with assertion. Evaluation of this expression should return TEXT content. This is typically a target url to which user is redirected after the recipient validates SAML token

signatureAlg

Algorithm to be used to sign/verify SAML transactions

Possible values: RSA-SHA1, RSA-SHA256

Default value: RSA-SHA1

digestMethod

Algorithm to be used to compute/verify digest for SAML transactions

Possible values: SHA1, SHA256

Default value: SHA1

audience

Audience for which assertion sent by IdP is applicable. This is typically entity name or url that represents ServiceProvider

Maximum value: 256

NameIDFormat

Format of Name Identifier sent in Assertion.

Possible values: Unspecified, emailAddress, X509SubjectName, WindowsDomainQualifiedName, kerberos, entity, persistent, transient

Default value: transient

NameIDExpr

Expression that will be evaluated to obtain NameIdentifier to be sent in assertion

Attribute1

Name of attribute1 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute1Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute1FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute1Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute2

Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute2Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute2FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute2Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute3

Name of attribute3 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute3Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute3FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute3Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute4

Name of attribute4 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute4Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute4FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute4Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute5

Name of attribute5 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute5Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute5FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute5Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute6

Name of attribute6 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute6Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute6FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute6Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute7

Name of attribute7 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute7Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute7FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute7Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute8

Name of attribute8 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute8Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute8FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute8Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute9

Name of attribute9 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute9Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute9FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute9Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute10

Name of attribute10 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute10Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute10FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute10Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute11

Name of attribute11 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute11Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute11FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute11Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute12

Name of attribute12 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute12Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute12FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute12Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute13

Name of attribute13 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute13Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute13FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute13Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute14

Name of attribute14 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute14Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute14FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute14Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute15

Name of attribute15 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute15Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute15FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute15Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

Attribute16

Name of attribute16 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute16Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute16FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Maximum value: 64

Attribute16Format

Format of Attribute1 to be sent in Assertion.

Possible values: URI, Basic

Default value: SAML_ATTR_UNSPECIFIED

unset vpn samlSSOProfile

Use this command to remove vpn samlSSOProfile settings. Refer to the set vpn samlSSOProfile command for meanings of the arguments.

Synopsys

```
unset vpn samlSSOProfile <name> [-samlSigningCertName] [-sendPassword] [-samlIssuerName] [-signatureAlg] [-digestMethod] [-audience] [-NameIDFormat] [-NameIDExpr] [-Attribute1FriendlyName] [-Attribute1Format] [-Attribute2FriendlyName] [-Attribute2Format] [-Attribute3FriendlyName] [-Attribute3Format] [-Attribute4FriendlyName] [-Attribute4Format] [-Attribute5FriendlyName] [-Attribute5Format] [-Attribute6FriendlyName] [-Attribute6Format] [-Attribute7FriendlyName] [-Attribute7Format] [-Attribute8FriendlyName] [-Attribute8Format] [-Attribute9FriendlyName] [-Attribute9Format] [-Attribute10FriendlyName] [-Attribute10Format] [-Attribute11FriendlyName] [-Attribute11Format] [-Attribute12FriendlyName] [-Attribute12Format] [-Attribute13FriendlyName] [-Attribute13Format] [-Attribute14FriendlyName] [-Attribute14Format] [-Attribute15FriendlyName] [-Attribute15Format] [-Attribute16FriendlyName] [-Attribute16Format]
```

show vpn samlSSOProfile

Displays information about all configured saml single sign-on profiles, or displays detailed information about the specified action.

Synopsys

```
show vpn samlSSOProfile [<name>]
```

Arguments

name

Name for the new saml single sign-on profile. Must begin with an ASCII alphanumeric or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after an SSO action is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my action" or 'my action').

Outputs

samlSigningCertName

Name of the signing authority as given in the SAML server's SSL certificate.

assertionConsumerServiceURL

URL to which the assertion is to be sent.

sendPassword

Option to send password in assertion.

samlIssuerName

The name to be used in requests sent from Netscaler to IdP to uniquely identify Netscaler.

relaystateRule

Expression to extract relaystate to be sent along with assertion. Evaluation of this expression should return TEXT content. This is typically a target url to which user is redirected after the recipient validates SAML token

signatureAlg

Algorithm to be used to sign/verify SAML transactions

digestMethod

Algorithm to be used to compute/verify digest for SAML transactions

audience

Audience for which assertion sent by IdP is applicable. This is typically entity name or url that represents ServiceProvider

NameIDFormat

Format of Name Identifier sent in Assertion.

NameIDExpr

Expression that will be evaluated to obtain NameIdentifier to be sent in assertion

Attribute1

Name of attribute1 that needs to be sent in SAML Assertion

Attribute2

Name of attribute2 that needs to be sent in SAML Assertion

Attribute3

Name of attribute3 that needs to be sent in SAML Assertion

Attribute4

Name of attribute4 that needs to be sent in SAML Assertion

Attribute5

Name of attribute5 that needs to be sent in SAML Assertion

Attribute6

Name of attribute6 that needs to be sent in SAML Assertion

Attribute7

Name of attribute7 that needs to be sent in SAML Assertion

Attribute8

Name of attribute8 that needs to be sent in SAML Assertion

Attribute9

Name of attribute9 that needs to be sent in SAML Assertion

Attribute10

Name of attribute10 that needs to be sent in SAML Assertion

Attribute11

Name of attribute11 that needs to be sent in SAML Assertion

Attribute12

Name of attribute12 that needs to be sent in SAML Assertion

Attribute13

Name of attribute13 that needs to be sent in SAML Assertion

Attribute14

Name of attribute14 that needs to be sent in SAML Assertion

Attribute15

Name of attribute15 that needs to be sent in SAML Assertion

Attribute16

Name of attribute16 that needs to be sent in SAML Assertion

Attribute1FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute2FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute3FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute4FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute5FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute6FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute7FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute8FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute9FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute10FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute11FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute12FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute13FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute14FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute15FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute16FriendlyName

User-Friendly Name of attribute2 that needs to be sent in SAML Assertion

Attribute1Format

Format of Attribute1 to be sent in Assertion.

Attribute2Format

Format of Attribute1 to be sent in Assertion.

Attribute3Format

Format of Attribute1 to be sent in Assertion.

Attribute4Format

Format of Attribute1 to be sent in Assertion.

Attribute5Format

Format of Attribute1 to be sent in Assertion.

Attribute6Format

Format of Attribute1 to be sent in Assertion.

Attribute7Format

Format of Attribute1 to be sent in Assertion.

Attribute8Format

Format of Attribute1 to be sent in Assertion.

Attribute9Format

Format of Attribute1 to be sent in Assertion.

Attribute10Format

Format of Attribute1 to be sent in Assertion.

Attribute11Format

Format of Attribute1 to be sent in Assertion.

Attribute12Format

Format of Attribute1 to be sent in Assertion.

Attribute13Format

Format of Attribute1 to be sent in Assertion.

Attribute14Format

Format of Attribute1 to be sent in Assertion.

Attribute15Format

Format of Attribute1 to be sent in Assertion.

Attribute16Format

Format of Attribute1 to be sent in Assertion.

Attribute1Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute2Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute3Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute4Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute5Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute6Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute7Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute8Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute9Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute10Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute11Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute12Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute13Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute14Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute15Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

Attribute16Expr

Expression that will be evaluated to obtain attribute1's value to be sent in Assertion

devno

count

stateflag

vpn sessionAction

The following operations can be performed on "vpn sessionAction":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add vpn sessionAction

Adds a session profile (action) to bind to a session policy that is applied to a user session if the policy expression conditions are met.

Synopsys

```
add vpn sessionAction <name> [-userAccounting <string>] [-httpPort <port> ...] [-winsIP <ip_addr>] [-
dnsVserverName <string>] [-splitDns <splitDns>] [-sessTimeout <mins>] [-clientSecurity <expression> [-
clientSecurityGroup <string>] [-clientSecurityMessage <string>]] [-clientSecurityLog ( ON | OFF )] [-splitTunnel
<splitTunnel>] [-localLanAccess ( ON | OFF )] [-rfc1918 ( ON | OFF )] [-killConnections ( ON | OFF )] [-
transparentInterception ( ON | OFF )] [-defaultAuthorizationAction ( ALLOW | DENY )] [-authorizationGroup <string>]
[-clientIdleTimeout <mins>] [-proxy <proxy>] [-allProtocolProxy <string>] [-httpProxy <string>] [-ftpProxy <string>] [-
socksProxy <string>] [-gopherProxy <string>] [-sslProxy <string>] [-proxyException <string>] [-proxyLocalBypass (
ENABLED | DISABLED )] [-clientCleanupPrompt ( ON | OFF )] [-forceCleanup <forceCleanup> ...] [-clientOptions
<clientOptions> ...] [-clientConfiguration <clientConfiguration> ...] [-SSO ( ON | OFF )] [-ssoCredential ( PRIMARY |
SECONDARY )] [-windowsAutoLogon ( ON | OFF )] [-useMIP ( NS | OFF )] [-useIIP <useIIP>] [-clientDebug
<clientDebug>] [-loginScript <input_filename>] [-logoutScript <input_filename>] [-homePage <URL>] [-icaProxy ( ON
| OFF )] [-wihome <URL>] [-wihomeAddressType ( IPV4 | IPV6 )] [-citrixReceiverHome <URL>] [-wiPortalMode (
NORMAL | COMPACT )] [-ClientChoices ( ON | OFF )] [-iipDnsSuffix <string>] [-forcedTimeout <mins>] [-
forcedTimeoutWarning <mins>] [-ntDomain <string>] [-clientlessVpnMode <clientlessVpnMode>] [-emailHome
<URL>] [-clientlessModeUrlEncoding <clientlessModeUrlEncoding>] [-clientlessPersistentCookie
<clientlessPersistentCookie>] [-allowedLoginGroups <string>] [-SecureBrowse ( ENABLED | DISABLED )] [-
storefronturl <string>] [-kcdAccount <string>]
```

Arguments

name

Name for the NetScaler Gateway profile (action). Must begin with an ASCII alphabetic or underscore (_) character, and must consist only of ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the profile is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my action" or 'my action').

userAccounting

The name of the radiusPolicy to use for RADIUS user accounting info on the session.

httpPort

Destination port numbers other than port 80, added as a comma-separated list. Traffic to these ports is processed as HTTP traffic, which allows functionality, such as HTTP authorization and single sign-on to a web application to work.

Minimum value: 1

winsIP

WINS server IP address to add to NetScaler Gateway for name resolution.

dnsVserverName

Name of the DNS virtual server for the user session.

splitDns

Route the DNS requests to the local DNS server configured on the user device, or NetScaler Gateway (remote), or both.

Possible values: LOCAL, REMOTE, BOTH

sessTimeout

Number of minutes after which the session times out.

Minimum value: 1

clientSecurity

Specify the client security check for the user device to permit a NetScaler Gateway session. The web address or IP address is not included in the expression for the client security check.

clientSecurityGroup

The client security group that will be assigned on failure of the client security check. Users can in general be organized into Groups. In this case, the Client Security Group may have a more restrictive security policy.

clientSecurityMessage

The client security message that will be displayed on failure of the client security check.

clientSecurityLog

Set the logging of client security checks.

Possible values: ON, OFF

splitTunnel

Send, through the tunnel, traffic only for intranet applications that are defined in NetScaler Gateway. Route all other traffic directly to the Internet. The OFF setting routes all traffic through NetScaler Gateway. With the REVERSE setting, intranet applications define the network traffic that is not intercepted. All network traffic directed to internal IP addresses bypasses the VPN tunnel, while other traffic goes through NetScaler Gateway. Reverse split tunneling can be used to log all non-local LAN traffic. For example, if users have a home network and are logged on through the NetScaler Gateway Plug-in, network traffic destined to a printer or another device within the home network is not intercepted.

Possible values: ON, OFF, REVERSE

localLanAccess

Set local LAN access. If split tunneling is OFF, and you set local LAN access to ON, the local client can route traffic to its local interface. When the local area network switch is specified, this combination of switches is useful. The client can allow local LAN access to devices that commonly have non-routable addresses, such as local printers or local file servers.

Possible values: ON, OFF

rfc1918

As defined in the local area network, allow only the following local area network addresses to bypass the VPN tunnel when the local LAN access feature is enabled:

* 10.*.*,*

* 172.16.*.*,*

* 192.168.*.*

Possible values: ON, OFF

killConnections

Specify whether the NetScaler Gateway Plug-in should disconnect all preexisting connections, such as the connections existing before the user logged on to NetScaler Gateway, and prevent new incoming connections on the NetScaler Gateway Plug-in for Windows and MAC when the user is connected to NetScaler Gateway and split tunneling is disabled.

Possible values: ON, OFF

transparentInterception

Allow access to network resources by using a single IP address and subnet mask or a range of IP addresses. The OFF setting sets the mode to proxy, in which you configure destination and source IP addresses and port numbers. If you are using the NetScaler Gateway Plug-in for Windows, set this parameter to ON, in which the mode is set to transparent. If you are using the NetScaler Gateway Plug-in for Java, set this parameter to OFF.

Possible values: ON, OFF

defaultAuthorizationAction

Specify the network resources that users have access to when they log on to the internal network. The default setting for authorization is to deny access to all network resources. Citrix recommends using the default global setting and then creating authorization policies to define the network resources users can access. If you set the default authorization policy to DENY, you must explicitly authorize access to any network resource, which improves security.

Possible values: ALLOW, DENY

authorizationGroup

Comma-separated list of groups in which the user is placed when none of the groups that the user is a part of is configured on NetScaler Gateway. The authorization policy can be bound to these groups to control access to the resources.

clientIdleTimeout

Time, in minutes, after which to time out the user session if NetScaler Gateway does not detect mouse or keyboard activity.

Minimum value: 1

Maximum value: 9999

proxy

Set options to apply proxy for accessing the internal resources. Available settings function as follows:

* BROWSER - Proxy settings are configured only in Internet Explorer and Firefox browsers.

* NS - Proxy settings are configured on the NetScaler appliance.

* OFF - Proxy settings are not configured.

Possible values: BROWSER, NS, OFF

allProtocolProxy

IP address of the proxy server to use for all protocols supported by NetScaler Gateway.

httpProxy

IP address of the proxy server to be used for HTTP access for all subsequent connections to the internal network.

ftpProxy

IP address of the proxy server to be used for FTP access for all subsequent connections to the internal network.

socksProxy

IP address of the proxy server to be used for SOCKS access for all subsequent connections to the internal network.

gopherProxy

IP address of the proxy server to be used for GOPHER access for all subsequent connections to the internal network.

sslProxy

IP address of the proxy server to be used for SSL access for all subsequent connections to the internal network.

proxyException

Proxy exception string that will be configured in the browser for bypassing the previously configured proxies. Allowed only if proxy type is Browser.

proxyLocalBypass

Bypass proxy server for local addresses option in Internet Explorer and Firefox proxy server settings.

Possible values: ENABLED, DISABLED

clientCleanupPrompt

Prompt for client-side cache clean-up when a client-initiated session closes.

Possible values: ON, OFF

forceCleanup

Force cache clean-up when the user closes a session. You can specify all, none, or any combination of the client-side items.

clientOptions

Display only the configured menu options when you select the "Configure NetScaler Gateway" option in the NetScaler Gateway Plug-in system tray icon for Windows.

clientConfiguration

Display only the configured tabs when you select the "Configure NetScaler Gateway" option in the NetScaler Gateway Plug-in system tray icon for Windows.

SSO

Set single sign-on (SSO) for the session. When the user accesses a server, the user's logon credentials are passed to the server for authentication.

Possible values: ON, OFF

ssoCredential

Specify whether to use the primary or secondary authentication credentials for single sign-on to the server.

Possible values: PRIMARY, SECONDARY

windowsAutoLogon

Enable or disable the Windows Auto Logon for the session. If a VPN session is established after this setting is enabled, the user is automatically logged on by using Windows credentials after the system is restarted.

Possible values: ON, OFF

useMIP

Enable or disable the use of a unique IP address alias, or a mapped IP address, as the client IP address for each client session. Allow NetScaler Gateway to use the mapped IP address as an intranet IP address when all other IP addresses are not available.

When IP pooling is configured and the mapped IP is used as an intranet IP address, the mapped IP address is used when an intranet IP address cannot be assigned.

Possible values: NS, OFF

useIIP

Define IP address pool options. Available settings function as follows:

* SPILLOVER - When an address pool is configured and the mapped IP is used as an intranet IP address, the mapped IP address is used when an intranet IP address cannot be assigned.

* NOSPILOVER - When intranet IP addresses are enabled and the mapped IP address is not used, the Transfer Login page appears for users who have used all available intranet IP addresses.

* OFF - Address pool is not configured.

Possible values: NOSPILOVER, SPILLOVER, OFF

clientDebug

Set the trace level on NetScaler Gateway. Technical support technicians use these debug logs for in-depth debugging and troubleshooting purposes. Available settings function as follows:

* DEBUG - Detailed debug messages are collected and written into the specified file.

* STATS - Application audit level error messages and debug statistic counters are written into the specified file.

* EVENTS - Application audit-level error messages are written into the specified file.

* OFF - Only critical events are logged into the Windows Application Log.

Possible values: debug, stats, events, OFF

loginScript

Path to the logon script that is run when a session is established. Separate multiple scripts by using comma. A "\$" in the path signifies that the word following the "\$" is an environment variable.

logoutScript

Path to the logout script. Separate multiple scripts by using comma. A "\$" in the path signifies that the word following the "\$" is an environment variable.

homePage

Web address of the home page that appears when users log on. Otherwise, users receive the default home page for NetScaler Gateway, which is the Access Interface.

icaProxy

Enable ICA proxy to configure secure Internet access to servers running Citrix XenApp or XenDesktop by using Citrix Receiver instead of the NetScaler Gateway Plug-in.

Possible values: ON, OFF

wihome

Web address of the Web Interface server, such as <http://<ipAddress>/Citrix/XenApp>, or Receiver for Web, which enumerates the virtualized resources, such as XenApp, XenDesktop, and cloud applications. This web address is used as the home page in ICA proxy mode.

If Client Choices is ON, you must configure this setting. Because the user can choose between FullClient and ICAProxy, the user may see a different home page. An Internet web site may appear if the user gets the FullClient option, or a Web Interface site if the user gets the ICAProxy option. If the setting is not configured, the XenApp option does not appear as a client choice.

wihomeAddressType

Type of the wihome address(IPV4/V6)

Possible values: IPV4, IPV6

citrixReceiverHome

Web address for the Citrix Receiver home page. Configure NetScaler Gateway so that when users log on to the appliance, the NetScaler Gateway Plug-in opens a web browser that allows single sign-on to the Citrix Receiver home page.

wiPortalMode

Layout on the Access Interface. The COMPACT value indicates the use of small icons.

Possible values: NORMAL, COMPACT

ClientChoices

Provide users with multiple logon options. With client choices, users have the option of logging on by using the NetScaler Gateway Plug-in for Windows, NetScaler Gateway Plug-in for Java, the Web Interface, or clientless access from one location. Depending on how NetScaler Gateway is configured, users are presented with up to three icons for logon choices. The most common are the NetScaler Gateway Plug-in for Windows, Web Interface, and clientless access.

Possible values: ON, OFF

iipDnsSuffix

An intranet IP DNS suffix. When a user logs on to NetScaler Gateway and is assigned an IP address, a DNS record for the user name and IP address combination is added to the NetScaler Gateway DNS cache. You can configure a DNS suffix to append to the user name when the DNS record is added to the cache. You can reach to the host from where the user is logged on by using the user's name, which can be easier to remember than an IP address. When the user logs off from NetScaler Gateway, the record is removed from the DNS cache.

forcedTimeout

Force a disconnection from the NetScaler Gateway Plug-in with NetScaler Gateway after a specified number of minutes. If the session closes, the user must log on again.

Minimum value: 1

Maximum value: 65535

forcedTimeoutWarning

Number of minutes to warn a user before the user session is disconnected.

Minimum value: 1

Maximum value: 255

ntDomain

Single sign-on domain to use for single sign-on to applications in the internal network. This setting can be overwritten by the domain that users specify at the time of logon or by the domain that the authentication server returns.

clientlessVpnMode

Enable clientless access for web, XenApp or XenDesktop, and FileShare resources without installing the NetScaler Gateway Plug-in. Available settings function as follows:

* ON - Allow only clientless access.

* OFF - Allow clientless access after users log on with the NetScaler Gateway Plug-in.

* DISABLED - Do not allow clientless access.

Possible values: ON, OFF, DISABLED

emailHome

Web address for the web-based email, such as Outlook Web Access.

clientlessModeUrlEncoding

When clientless access is enabled, you can choose to encode the addresses of internal web applications or to leave the address as clear text. Available settings function as follows:

* OPAQUE - Use standard encoding mechanisms to make the domain and protocol part of the resource unclear to users.

* CLEAR - Do not encode the web address and make it visible to users.

* ENCRYPT - Allow the domain and protocol to be encrypted using a session key. When the web address is encrypted, the URL is different for each user session for the same web resource. If users bookmark the encoded web address, save it in the web browser and then log off, they cannot connect to the web address when they log on and use the bookmark. If users save the encrypted bookmark in the Access Interface during their session, the bookmark works each time the user logs on.

Possible values: TRANSPARENT, OPAQUE, ENCRYPT

clientlessPersistentCookie

State of persistent cookies in clientless access mode. Persistent cookies are required for accessing certain features of SharePoint, such as opening and editing Microsoft Word, Excel, and PowerPoint documents hosted on the SharePoint server. A persistent cookie remains on the user device and is sent with each HTTP request. NetScaler Gateway encrypts the persistent cookie before sending it to the plug-in on the user device, and refreshes the cookie periodically as long as the session exists. The cookie becomes stale if the session ends. Available settings function as follows:

* ALLOW - Enable persistent cookies. Users can open and edit Microsoft documents stored in SharePoint.

* DENY - Disable persistent cookies. Users cannot open and edit Microsoft documents stored in SharePoint.

* PROMPT - Prompt users to allow or deny persistent cookies during the session. Persistent cookies are not required for clientless access if users do not connect to SharePoint.

Possible values: ALLOW, DENY, PROMPT

allowedLoginGroups

Specify groups that have permission to log on to NetScaler Gateway. Users who do not belong to this group or groups are denied access even if they have valid credentials.

SecureBrowse

Allow users to connect through NetScaler Gateway to network resources from iOS and Android mobile devices with Citrix Receiver. Users do not need to establish a full VPN tunnel to access resources in the secure network.

Possible values: ENABLED, DISABLED

storefronturl

Web address for StoreFront to be used in this session for enumeration of resources from XenApp or XenDesktop.

kcdAccount

The kcd account details to be used in SSO

rm vpn sessionAction

Removes an action that was previously added to a session policy.

Synopsis

rm vpn sessionAction <name>

Arguments

name

Name of the action to remove.

set vpn sessionAction

Modifies an action that was previously added to a session policy that is applied to a user session if the policy expression conditions are met.

Synopsys

```
set vpn sessionAction <name> [-userAccounting <string>] [-httpPort <port> ...] [-winsIP <ip_addr>] [-  
dnsVserverName <string>] [-splitDns <splitDns>] [-sessTimeout <mins>] [-clientSecurity <expression> [-  
clientSecurityGroup <string>] [-clientSecurityMessage <string>]] [-clientSecurityLog ( ON | OFF )] [-splitTunnel  
<splitTunnel>] [-localLanAccess ( ON | OFF )] [-rfc1918 ( ON | OFF )] [-killConnections ( ON | OFF )] [-  
transparentInterception ( ON | OFF )] [-defaultAuthorizationAction ( ALLOW | DENY )] [-authorizationGroup <string>]  
[-clientIdleTimeout <mins>] [-proxy <proxy>] [-allProtocolProxy <string> | -httpProxy <string> | -ftpProxy <string> | -  
socksProxy <string> | -gopherProxy <string> | -sslProxy <string>] [-proxyException <string>] [-proxyLocalBypass (   
ENABLED | DISABLED )] [-clientCleanupPrompt ( ON | OFF )] [-forceCleanup <forceCleanup> ...] [-clientOptions  
<clientOptions> ...] [-clientConfiguration <clientConfiguration> ...] [-SSO ( ON | OFF )] [-ssoCredential ( PRIMARY |  
SECONDARY )] [-windowsAutoLogon ( ON | OFF )] [-useMIP ( NS | OFF )] [-useIIP <useIIP>] [-clientDebug  
<clientDebug>] [-loginScript <input_filename>] [-logoutScript <input_filename>] [-homePage <URL>] [-icaProxy ( ON  
| OFF )] [-wihome <URL>] [-wihomeAddressType ( IPV4 | IPV6 )] [-citrixReceiverHome <URL>] [-wiPortalMode (   
NORMAL | COMPACT )] [-ClientChoices ( ON | OFF )] [-iipDnsSuffix <string>] [-forcedTimeout <mins>] [-  
forcedTimeoutWarning <mins>] [-ntDomain <string>] [-clientlessVpnMode <clientlessVpnMode>] [-emailHome  
<URL>] [-clientlessModeUrlEncoding <clientlessModeUrlEncoding>] [-clientlessPersistentCookie  
<clientlessPersistentCookie>] [-allowedLoginGroups <string>] [-SecureBrowse ( ENABLED | DISABLED )] [-  
storefronturl <string>] [-kcdAccount <string>]
```

Arguments

name

The name of the vpn session action.

userAccounting

Name of RADIUS Policy to use for user accounting

httpPort

Destination port numbers other than port 80, added as a comma-separated list. Traffic to these ports is processed as HTTP traffic, which allows functionality, such as HTTP authorization and single sign-on to a web application to work.

Minimum value: 1

winsIP

The WINS server ip address.

dnsVserverName

Name of the DNS virtual server for the user session.

splitDns

Route the DNS requests to the local DNS server configured on the user device, or NetScaler Gateway (remote), or both.

Possible values: LOCAL, REMOTE, BOTH

sessTimeout

Number of minutes after which the session times out.

Minimum value: 1

clientSecurity

Specify the client security check for the user device to permit a NetScaler Gateway session. The web address or IP address is not included in the expression for the client security check.

clientSecurityGroup

The client security group that will be assigned on failure of the client security check. Users can in general be organized into Groups. In this case, the Client Security Group may have a more restrictive security policy.

clientSecurityMessage

The client security message that will be displayed on failure of the client security check.

clientSecurityLog

Set the logging of client security checks.

Possible values: ON, OFF

splitTunnel

Send, through the tunnel, traffic only for intranet applications that are defined in NetScaler Gateway. Route all other traffic directly to the Internet. The OFF setting routes all traffic through NetScaler Gateway. With the REVERSE setting, intranet applications define the network traffic that is not intercepted. All network traffic directed to internal IP addresses bypasses the VPN tunnel, while other traffic goes through NetScaler Gateway. Reverse split tunneling can be used to log all non-local LAN traffic. For example, if users have a home network and are logged on through the NetScaler Gateway Plug-in, network traffic destined to a printer or another device within the home network is not intercepted.

Possible values: ON, OFF, REVERSE

localLanAccess

Set local LAN access. If split tunneling is OFF, and you set local LAN access to ON, the local client can route traffic to its local interface. When the local area network switch is specified, this combination of switches is useful. The client can allow local LAN access to devices that commonly have non-routable addresses, such as local printers or local file servers.

Possible values: ON, OFF

rfc1918

As defined in the local area network, allow only the following local area network addresses to bypass the VPN tunnel when the local LAN access feature is enabled:

* 10.*.*.,

* 172.16.*.*,

* 192.168.*.*

Possible values: ON, OFF

killConnections

Specify whether the NetScaler Gateway Plug-in should disconnect all preexisting connections, such as the connections existing before the user logged on to NetScaler Gateway, and prevent new incoming connections on the NetScaler Gateway Plug-in for Windows and MAC when the user is connected to NetScaler Gateway and split tunneling is disabled.

Possible values: ON, OFF

transparentInterception

Allow access to network resources by using a single IP address and subnet mask or a range of IP addresses. The OFF setting sets the mode to proxy, in which you configure destination and source IP addresses and port numbers. If you are using the NetScaler Gateway Plug-in for Windows, set this parameter to ON, in which the mode is set to transparent. If you are using the NetScaler Gateway Plug-in for Java, set this parameter to OFF.

Possible values: ON, OFF

defaultAuthorizationAction

Specify the network resources that users have access to when they log on to the internal network. The default setting for authorization is to deny access to all network resources. Citrix recommends using the default global setting and then creating authorization policies to define the network resources users can access. If you set the default authorization policy to DENY, you must explicitly authorize access to any network resource, which improves security.

Possible values: ALLOW, DENY

authorizationGroup

Comma-separated list of groups in which the user is placed when none of the groups that the user is a part of is configured on NetScaler Gateway. The authorization policy can be bound to these groups to control access to the resources.

clientIdleTimeout

Time, in minutes, after which to time out the user session if NetScaler Gateway does not detect mouse or keyboard activity.

Minimum value: 1

Maximum value: 9999

proxy

Set options to apply proxy for accessing the internal resources. Available settings function as follows:

* BROWSER - Proxy settings are configured only in Internet Explorer and Firefox browsers.

* NS - Proxy settings are configured on the NetScaler appliance.

* OFF - Proxy settings are not configured.

Possible values: BROWSER, NS, OFF

allProtocolProxy

IP address of the proxy server to use for all protocols supported by NetScaler Gateway.

httpProxy

IP address of the proxy server to be used for HTTP access for all subsequent connections to the internal network.

ftpProxy

IP address of the proxy server to be used for FTP access for all subsequent connections to the internal network.

socksProxy

IP address of the proxy server to be used for SOCKS access for all subsequent connections to the internal network.

gopherProxy

IP address of the proxy server to be used for GOPHER access for all subsequent connections to the internal network.

sslProxy

IP address of the proxy server to be used for SSL access for all subsequent connections to the internal network.

proxyException

Proxy exception string that will be configured in the browser for bypassing the previously configured proxies. Allowed only if proxy type is Browser.

proxyLocalBypass

Bypass proxy server for local addresses option in Internet Explorer and Firefox proxy server settings.

Possible values: ENABLED, DISABLED

clientCleanupPrompt

Prompt for client-side cache clean-up when a client-initiated session closes.

Possible values: ON, OFF

forceCleanup

Force cache clean-up when the user closes a session. You can specify all, none, or any combination of the client-side items.

clientOptions

Display only the configured menu options when you select the "Configure NetScaler Gateway" option in the NetScaler Gateway Plug-in system tray icon for Windows.

clientConfiguration

Display only the configured tabs when you select the "Configure NetScaler Gateway" option in the NetScaler Gateway Plug-in system tray icon for Windows.

SSO

Set single sign-on (SSO) for the session. When the user accesses a server, the user's logon credentials are passed to the server for authentication.

Possible values: ON, OFF

ssoCredential

Specify whether to use the primary or secondary authentication credentials for single sign-on to the server.

Possible values: PRIMARY, SECONDARY

windowsAutoLogon

Enable or disable the Windows Auto Logon for the session. If a VPN session is established after this setting is enabled, the user is automatically logged on by using Windows credentials after the system is restarted.

Possible values: ON, OFF

useMIP

Enable or disable the use of a unique IP address alias, or a mapped IP address, as the client IP address for each client session. Allow NetScaler Gateway to use the mapped IP address as an intranet IP address when all other IP addresses are not available.

When IP pooling is configured and the mapped IP is used as an intranet IP address, the mapped IP address is used when an intranet IP address cannot be assigned.

Possible values: NS, OFF

useIIP

Define IP address pool options. Available settings function as follows:

* SPILLOVER - When an address pool is configured and the mapped IP is used as an intranet IP address, the mapped IP address is used when an intranet IP address cannot be assigned.

* NOSPILOVER - When intranet IP addresses are enabled and the mapped IP address is not used, the Transfer Login page appears for users who have used all available intranet IP addresses.

* OFF - Address pool is not configured.

Possible values: NOSPILOVER, SPILLOVER, OFF

clientDebug

Set the trace level on NetScaler Gateway. Technical support technicians use these debug logs for in-depth debugging and troubleshooting purposes. Available settings function as follows:

- * **DEBUG** - Detailed debug messages are collected and written into the specified file.
- * **STATS** - Application audit level error messages and debug statistic counters are written into the specified file.
- * **EVENTS** - Application audit-level error messages are written into the specified file.
- * **OFF** - Only critical events are logged into the Windows Application Log.

Possible values: debug, stats, events, OFF

loginScript

Path to the logon script that is run when a session is established. Separate multiple scripts by using comma. A "\$" in the path signifies that the word following the "\$" is an environment variable.

logoutScript

Path to the logout script. Separate multiple scripts by using comma. A "\$" in the path signifies that the word following the "\$" is an environment variable.

homePage

Web address of the home page that appears when users log on. Otherwise, users receive the default home page for NetScaler Gateway, which is the Access Interface.

icaProxy

Enable ICA proxy to configure secure Internet access to servers running Citrix XenApp or XenDesktop by using Citrix Receiver instead of the NetScaler Gateway Plug-in.

Possible values: ON, OFF

Default value: OFF

wihome

Web address of the Web Interface server, such as `http://<ipAddress>/Citrix/XenApp`, or Receiver for Web, which enumerates the virtualized resources, such as XenApp, XenDesktop, and cloud applications. This web address is used as the home page in ICA proxy mode.

If Client Choices is ON, you must configure this setting. Because the user can choose between FullClient and ICAProxy, the user may see a different home page. An Internet web site may appear if the user gets the FullClient option, or a Web Interface site if the user gets the ICAProxy option. If the setting is not configured, the XenApp option does not appear as a client choice.

wihomeAddressType

Type of the wihome address(IPV4/V6)

Possible values: IPV4, IPV6

citrixReceiverHome

Web address for the Citrix Receiver home page. Configure NetScaler Gateway so that when users log on to the appliance, the NetScaler Gateway Plug-in opens a web browser that allows single sign-on to the Citrix Receiver home page.

wiPortalMode

Layout on the Access Interface. The COMPACT value indicates the use of small icons.

Possible values: NORMAL, COMPACT

ClientChoices

Provide users with multiple logon options. With client choices, users have the option of logging on by using the NetScaler Gateway Plug-in for Windows, NetScaler Gateway Plug-in for Java, the Web Interface, or clientless access from one location. Depending on how NetScaler Gateway is configured, users are presented with up to three icons for logon choices. The most common are the NetScaler Gateway Plug-in for Windows, Web Interface, and clientless access.

Possible values: ON, OFF

iipDnsSuffix

An intranet IP DNS suffix. When a user logs on to NetScaler Gateway and is assigned an IP address, a DNS record for the user name and IP address combination is added to the NetScaler Gateway DNS cache. You can configure a DNS suffix to append to the user name when the DNS record is added to the cache. You can reach to the host from where the user is logged on by using the user's name, which can be easier to remember than an IP address. When the user logs off from NetScaler Gateway, the record is removed from the DNS cache.

forcedTimeout

Force a disconnection from the NetScaler Gateway Plug-in with NetScaler Gateway after a specified number of minutes. If the session closes, the user must log on again.

Minimum value: 1

Maximum value: 65535

forcedTimeoutWarning

Number of minutes to warn a user before the user session is disconnected.

Minimum value: 1

Maximum value: 255

ntDomain

Single sign-on domain to use for single sign-on to applications in the internal network. This setting can be overwritten by the domain that users specify at the time of logon or by the domain that the authentication server returns.

clientlessVpnMode

Enable clientless access for web, XenApp or XenDesktop, and FileShare resources without installing the NetScaler Gateway Plug-in. Available settings function as follows:

* ON - Allow only clientless access.

* OFF - Allow clientless access after users log on with the NetScaler Gateway Plug-in.

* DISABLED - Do not allow clientless access.

Possible values: ON, OFF, DISABLED

Default value: OFF

emailHome

Web address for the web-based email, such as Outlook Web Access.

clientlessModeUrlEncoding

When clientless access is enabled, you can choose to encode the addresses of internal web applications or to leave the address as clear text. Available settings function as follows:

* OPAQUE - Use standard encoding mechanisms to make the domain and protocol part of the resource unclear to users.

* CLEAR - Do not encode the web address and make it visible to users.

* ENCRYPT - Allow the domain and protocol to be encrypted using a session key. When the web address is encrypted, the URL is different for each user session for the same web resource. If users bookmark the encoded web address, save it in the web browser and then log off, they cannot connect to the web address when they log on and use the bookmark. If users save the encrypted bookmark in the Access Interface during their session, the bookmark works each time the user logs on.

Possible values: TRANSPARENT, OPAQUE, ENCRYPT

clientlessPersistentCookie

State of persistent cookies in clientless access mode. Persistent cookies are required for accessing certain features of SharePoint, such as opening and editing Microsoft Word, Excel, and PowerPoint documents hosted on the SharePoint server. A persistent cookie remains on the user device and is sent with each HTTP request. NetScaler Gateway encrypts the persistent cookie before sending it to the plug-in on the user device, and refreshes the cookie periodically as long as the session exists. The cookie becomes stale if the session ends. Available settings function as follows:

* ALLOW - Enable persistent cookies. Users can open and edit Microsoft documents stored in SharePoint.

* DENY - Disable persistent cookies. Users cannot open and edit Microsoft documents stored in SharePoint.

* PROMPT - Prompt users to allow or deny persistent cookies during the session. Persistent cookies are not required for clientless access if users do not connect to SharePoint.

Possible values: ALLOW, DENY, PROMPT

Default value: DENY

allowedLoginGroups

Specify groups that have permission to log on to NetScaler Gateway. Users who do not belong to this group or groups are denied access even if they have valid credentials.

SecureBrowse

Allow users to connect through NetScaler Gateway to network resources from iOS and Android mobile devices with Citrix Receiver. Users do not need to establish a full VPN tunnel to access resources in the secure network.

Possible values: ENABLED, DISABLED

storefronturl

Web address for StoreFront to be used in this session for enumeration of resources from XenApp or XenDesktop.

kcdAccount

The kcd account details to be used in SSO

unset vpn sessionAction

Use this command to remove vpn sessionAction settings. Refer to the set vpn sessionAction command for meanings of the arguments.

Synopsys

```
unset vpn sessionAction <name> [-userAccounting] [-httpPort] [-winsIP] [-dnsVserverName] [-splitDns] [-sessTimeout] [-clientSecurity] [-clientSecurityGroup] [-clientSecurityMessage] [-clientSecurityLog] [-splitTunnel] [-localLanAccess] [-rfc1918] [-killConnections] [-transparentInterception] [-defaultAuthorizationAction] [-authorizationGroup] [-clientIdleTimeout] [-proxy] [-allProtocolProxy] [-httpProxy] [-ftpProxy] [-socksProxy] [-gopherProxy] [-sslProxy] [-proxyException] [-proxyLocalBypass] [-clientCleanupPrompt] [-forceCleanup] [-clientOptions] [-clientConfiguration] [-SSO] [-ssoCredential] [-windowsAutoLogon] [-useMIP] [-useIIP] [-clientDebug] [-loginScript] [-logoutScript] [-homePage] [-icaProxy] [-wihome] [-citrixReceiverHome] [-wiPortalMode] [-ClientChoices] [-iipDnsSuffix] [-forcedTimeout] [-forcedTimeoutWarning] [-ntDomain] [-clientlessVpnMode] [-emailHome] [-clientlessModeUrlEncoding] [-clientlessPersistentCookie] [-allowedLoginGroups] [-SecureBrowse] [-storefronturl] [-kcdAccount]
```

show vpn sessionAction

Displays a session action that is applied to a user session if the policy expression conditions are met.

Synopsys

show vpn sessionAction [<name>]

Arguments

name

Name of the session action to display.

Outputs

userAccounting

RADIUS policy to use for user accounting

httpPort

The HTTP port for this session action

winsIP

The WINS server IP address for this session action.

dnsVserverName

The name of the DNS vserver configured by the session action.

splitDns

The VPN client SplitDns state.

sessTimeout

The session timeout, in minutes, set by the action.

clientSecurity

The client security check string being applied. This is in the form of an expression. Expressions are simple conditions, such as a test for equality, applied to operands, such as a URL string or an IP address. Expression syntax is described in the Installation and Configuration Guide.

clientSecurityGroup

The client security group that will be assigned on failure of the client security check. Users can in general be organized into Groups. In this case, the Client Security Group may have a more restrictive security policy.

clientSecurityMessage

The client security message that will be displayed on failure of the client security check.

clientSecurityLog

Set the logging of client security checks.

splitTunnel

Send, through the tunnel, traffic only for intranet applications that are defined in NetScaler Gateway. Route all other traffic directly to the Internet. The OFF setting routes all traffic through NetScaler Gateway. With the REVERSE setting, intranet applications define the network traffic that is not intercepted. All network traffic directed to internal IP addresses bypasses the VPN tunnel, while other traffic goes through NetScaler Gateway. Reverse split tunneling can be used to log all non-local LAN traffic. For example, if users have a home network and are logged on through the NetScaler Gateway Plug-in, network traffic destined to a printer or another device within the home network is not intercepted.

localLanAccess

Set local LAN access. If split tunneling is OFF, and you set local LAN access to ON, the local client can route traffic to its local interface. When the local area network switch is specified, this combination of switches is useful. The client can allow local LAN access to devices that commonly have non-routable addresses, such as local printers or local file servers.

rfc1918

As defined in the local area network, allow only the following local area network addresses to bypass the VPN tunnel when the local LAN access feature is enabled:

* 10.*.*.,

* 172.16.*.*,

* 192.168.*.*

spoofIP

IP address that the intranet application uses to route the connection through the virtual adapter.

killConnections

Specify whether the NetScaler Gateway Plug-in should disconnect all preexisting connections, such as the connections existing before the user logged on to NetScaler Gateway, and prevent new incoming connections on the NetScaler Gateway Plug-in for Windows and MAC when the user is connected to NetScaler Gateway and split tunneling is disabled.

transparentInterception

Allow access to network resources by using a single IP address and subnet mask or a range of IP addresses. The OFF setting sets the mode to proxy, in which you configure destination and source IP addresses and port numbers. If you are using the NetScaler Gateway Plug-in for Windows, set this parameter to ON, in which the mode is set to transparent. If you are using the NetScaler Gateway Plug-in for Java, set this parameter to OFF.

windowsClientType

Windows client type, e.g. Agent or ActiveX

defaultAuthorizationAction

The Authorization Action, e.g. allow or deny

authorizationGroup

The authorization group applied to client sessions.

clientIdleTimeout

The client idle timeout, in minutes.

clientIdleTimeoutWarning

The time after which the client gets a timeout warning, in minutes.

proxy

The state of proxy configuration for the session.

allProtocolProxy

The address set for all proxies.

httpProxy

The HTTP proxy IP address.

ftpProxy

The FTP proxy IP address.

socksProxy

IP address of the proxy server to be used for SOCKS access for all subsequent connections to the internal network.

gopherProxy

The Gopher proxy IP address.

sslProxy

The HTTPS proxy IP address.

proxyException

Proxy exception string that will be configured in the browser for bypassing the previously configured proxies. Allowed only if proxy type is Browser.

proxyLocalBypass

Bypass proxy server for local addresses option in Internet Explorer and Firefox proxy server settings.

clientCleanupPrompt

Prompt for client-side cache clean-up when a client-initiated session closes.

forceCleanup

Force cache clean-up when the user closes a session. You can specify all, none, or any combination of the client-side items.

clientOptions

List of configured buttons(and/or menu options in the docked client) in the Windows VPN client.

clientConfiguration

List of configured tabs in the Windows VPN client.

SSO

Whether or not Single Sign-On is used for this session.

ssoCredential

Specify whether to use the primary or secondary authentication credentials for single sign-on to the server.

windowsAutoLogon

Whether or not Windows Auto Logon is enabled for this session.

useMIP

Whether or not a Mapped IP address is used for the session

useIIP

Define IP address pool options. Available settings function as follows:

- * SPILLOVER - When an address pool is configured and the mapped IP is used as an intranet IP address, the mapped IP address is used when an intranet IP address cannot be assigned.

- * NOSPILOVER - When intranet IP addresses are enabled and the mapped IP address is not used, the Transfer Login page appears for users who have used all available intranet IP addresses.

- * OFF - Address pool is not configured.

clientDebug

Trace level on the Windows VPN Client.

loginScript

Path to the logon script that is run when a session is established. Separate multiple scripts by using comma. A "\$" in the path signifies that the word following the "\$" is an environment variable.

logoutScript

Path to the logout script. Separate multiple scripts by using comma. A "\$" in the path signifies that the word following the "\$" is an environment variable.

homePage

The client home page.

icaProxy

Enable ICA proxy to configure secure Internet access to servers running Citrix XenApp or XenDesktop by using Citrix Receiver instead of the NetScaler Gateway Plug-in.

wihome

Web address of the Web Interface server, such as `http://<ipAddress>/Citrix/XenApp`, or Receiver for Web, which enumerates the virtualized resources, such as XenApp, XenDesktop, and cloud applications. This web address is used as the home page in ICA proxy mode.

If Client Choices is ON, you must configure this setting. Because the user can choose between FullClient and ICAProxy, the user may see a different home page. An Internet web site may appear if the user gets the FullClient option, or a Web Interface site if the user gets the ICAProxy option. If the setting is not configured, the XenApp option does not appear as a client choice.

wihomeAddressType

Type of the wihome address(IPV4/V6)

citrixReceiverHome

Web address for the Citrix Receiver home page. Configure NetScaler Gateway so that when users log on to the appliance, the NetScaler Gateway Plug-in opens a web browser that allows single sign-on to the Citrix Receiver home page.

wiPortalMode

Layout on the Access Interface. The COMPACT value indicates the use of small icons.

ClientChoices

Provide users with multiple logon options. With client choices, users have the option of logging on by using the NetScaler Gateway Plug-in for Windows, NetScaler Gateway Plug-in for Java, the Web Interface, or clientless access from one location. Depending on how NetScaler Gateway is configured, users are presented with up to three icons for logon choices. The most common are the NetScaler Gateway Plug-in for Windows, Web Interface, and clientless access.

epaClientType

Choose between two types of End point Windows Client

- a) Application Agent - which always runs in the task bar as a standalone application and also has a supporting service which runs permanently when installed
- b) Activex Control - ActiveX control run by Microsoft Internet Explorer.

iipDnsSuffix

The IntranetIP DNS suffix.

forcedTimeout

Force a disconnection from the NetScaler Gateway Plug-in with NetScaler Gateway after a specified number of minutes. If the session closes, the user must log on again.

forcedTimeoutWarning

Number of minutes to warn a user before the user session is disconnected.

ntDomain

Single sign-on domain to use for single sign-on to applications in the internal network. This setting can be overwritten by the domain that users specify at the time of logon or by the domain that the authentication server returns.

clientlessVpnMode

Whether clientlessVPN is available to the session.

clientlessModeUrlEncoding

URL encoding used in clientless mode.

clientlessPersistentCookie

State of persistent cookies in clientless access mode. Persistent cookies are required for accessing certain features of SharePoint, such as opening and editing Microsoft Word, Excel, and PowerPoint documents hosted on the SharePoint server. A persistent cookie remains on the user device and is sent with each HTTP request. NetScaler Gateway encrypts the persistent cookie before sending it to the plug-in on the user device, and refreshes the cookie periodically as long as the session exists. The cookie becomes stale if the session ends. Available settings function as follows:

- * ALLOW - Enable persistent cookies. Users can open and edit Microsoft documents stored in SharePoint.
- * DENY - Disable persistent cookies. Users cannot open and edit Microsoft documents stored in SharePoint.
- * PROMPT - Prompt users to allow or deny persistent cookies during the session. Persistent cookies are not required for clientless access if users do not connect to SharePoint.

emailHome

The EMail home for the portal

stateflag

allowedLoginGroups

Specify groups that have permission to log on to NetScaler Gateway. Users who do not belong to this group or groups are denied access even if they have valid credentials.

SecureBrowse

Allow users to connect through NetScaler Gateway to network resources from iOS and Android mobile devices with Citrix Receiver. Users do not need to establish a full VPN tunnel to access resources in the secure network.

storefronturl

Web address for StoreFront to be used in this session for enumeration of resources from XenApp or XenDesktop.

kcdAccount

The kcd account details to be used in SSO

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

devno

count

vpn sessionPolicy

The following operations can be performed on "vpn sessionPolicy":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add vpn sessionPolicy

Creates a new session policy that, if bound, is applied after the user logs on to NetScaler Gateway, and that determines the properties of the user session.

Synopsys

```
add vpn sessionPolicy <name> <rule> <action>
```

Arguments

name

Name for the new session policy that is applied after the user logs on to NetScaler Gateway.

rule

Expression, or name of a named expression, specifying the traffic that matches the policy. Can be written in either default or classic syntax.

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

The following requirements apply only to the NetScaler CLI:

- * If the expression includes one or more spaces, enclose the entire expression in double quotation marks.
- * If the expression itself includes double quotation marks, escape the quotations by using the \ character.
- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Action to be applied by the new session policy if the rule criteria are met.

rm vpn sessionPolicy

Removes the session policy that is applied after the user logs on to NetScaler Gateway.

Synopsys

```
rm vpn sessionPolicy <name>
```

Arguments

name

Name of the session policy to remove.

set vpn sessionPolicy

Modifies the rule or action of a session policy.

Synopsys

```
set vpn sessionPolicy <name> [-rule <expression>] [-action <string>]
```

Arguments

name

Name of the session policy to modify.

rule

Expression, or name of a named expression, specifying the traffic that matches the policy. Can be written in either default or classic syntax.

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters> + <string of 245 characters>"

The following requirements apply only to the NetScaler CLI:

- * If the expression includes one or more spaces, enclose the entire expression in double quotation marks.
- * If the expression itself includes double quotation marks, escape the quotations by using the \ character.
- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Action to be applied by the new session policy if the rule criteria are met.

unset vpn sessionPolicy

Use this command to remove vpn sessionPolicy settings. Refer to the set vpn sessionPolicy command for meanings of the arguments.

Synopsis

```
unset vpn sessionPolicy <name> [-rule] [-action]
```

show vpn sessionPolicy

Displays a session policy.

Synopsis

```
show vpn sessionPolicy [<name>]
```

Arguments

name

Name of the session policy to display.

Outputs

rule

The new rule associated with the policy. Rules are combinations of expressions. Expressions are simple conditions, such as a test for equality, applied to operands, such as a URL string or an IP address. Expression syntax is described in the Installation and Configuration Guide.

action

The new vpn session action the policy is using.

boundTo

The entity name to which policy is bound

activePolicy

priority

bindPolicyType

policyType

builtin

Indicates that a variable is a built-in (SYSTEM INTERNAL) type.

devno

count

stateflag

vpn stats

The following operations can be performed on "vpn stats":

show vpn stats

show vpn stats is an alias for stat vpn

Synopsys

show vpn stats - alias for 'stat vpn'

vpn trafficAction

The following operations can be performed on "vpn trafficAction":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add vpn trafficAction

Creates an action to be applied by a policy that matches the traffic being processed.

Synopsys

```
add vpn trafficAction <name> <qual> [-appTimeout <mins>] [(-SSO ( ON | OFF ) [-formSSOAction <string>]) | -wanscaler ( ON | OFF )] [-HDX ( ON | OFF )] [-fta ( ON | OFF )] [-kcdAccount <string>] [-samlSSOProfile <string>] [-proxy <string>] [-userExpression <string>] [-passwdExpression <string>]
```

Arguments

name

Name for the traffic action. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after a traffic action is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my action" or 'my action').

qual

Protocol, either HTTP or TCP, to be used with the action.

Possible values: http, tcp

appTimeout

Maximum amount of time, in minutes, a user can stay logged on to the web application.

Minimum value: 1

Maximum value: 715827

SSO

Provide single sign-on to the web application.

Possible values: ON, OFF

HDX

Provide hdx proxy to the ICA traffic

Possible values: ON, OFF

formSSOAction

Name of the form-based single sign-on profile. Form-based single sign-on allows users to log on one time to all protected applications in your network, instead of requiring them to log on separately to access each one.

fta

Specify file type association, which is a list of file extensions that users are allowed to open.

Possible values: ON, OFF

wanscaler

Use the Repeater Plug-in to optimize network traffic.

Possible values: ON, OFF

kcdAccount

Kerberos constrained delegation account name

Default value: "Default"

samlSSOProfile

Profile to be used for doing SAML SSO to remote relying party

proxy

IP address and Port of the proxy server to be used for HTTP access for this request.

userExpression

expression that will be evaluated to obtain username for SingleSignOn

Maximum value: 256

passwdExpression

expression that will be evaluated to obtain password for SingleSignOn

Maximum value: 256

rm vpn trafficAction

Removes a previously created traffic policy action.

Synopsys

rm vpn trafficAction <name>

Arguments

name

Name of the traffic policy action to remove.

set vpn trafficAction

Modifies a traffic policy action to be applied by the policy if the rule criteria are met.

Synopsys

set vpn trafficAction <name> [-appTimeout <mins>] [-SSO (ON | OFF)] [-wanscaler (ON | OFF)] [-HDX (ON | OFF)] [-formSSOAction <string>] [-fta (ON | OFF)] [-kcdAccount <string>] [-samlSSOProfile <string>] [-proxy <string>] [-userExpression <string>] [-passwdExpression <string>]

Arguments

name

Name of the traffic policy action to modify.

appTimeout

Maximum amount of time, in minutes, a user can stay logged on to the web application.

Minimum value: 1

Maximum value: 715827

SSO

Provide single sign-on to the web application.

Possible values: ON, OFF

HDX

Provide hdx proxy to the ICA traffic

Possible values: ON, OFF

formSSOAction

Name of the form-based single sign-on profile. Form-based single sign-on allows users to log on one time to all protected applications in your network, instead of requiring them to log on separately to access each one.

fta

Specify file type association, which is a list of file extensions that users are allowed to open.

Possible values: ON, OFF

wanscaler

Use the Repeater Plug-in to optimize network traffic.

Possible values: ON, OFF

kcdAccount

Kerberos constrained delegation account name

Default value: "Default"

samlSSOProfile

Profile to be used for doing SAML SSO to remote relying party

proxy

IP address and Port of the proxy server to be used for HTTP access for this request.

userExpression

expression that will be evaluated to obtain username for SingleSignOn

Maximum value: 256

passwdExpression

expression that will be evaluated to obtain password for SingleSignOn

Maximum value: 256

unset vpn trafficAction

Use this command to remove vpn trafficAction settings. Refer to the set vpn trafficAction command for meanings of the arguments.

Synopsys

```
unset vpn trafficAction <name> [-wanscaler] [-kcdAccount] [-proxy] [-userExpression] [-passwdExpression]
```

show vpn trafficAction

Displays information about all the configured traffic actions, or displays detailed information about the specified traffic action.

Synopsys

show vpn trafficAction [<name>]

Arguments

name

Name of the traffic policy action for which to display detailed information.

Outputs

qual

The protocol that is set with the action; for example, http or tcp.

appTimeout

The application timeout

SSO

Whether or not Single Sign On is enabled.

formSSOAction

Name of the form-based single sign-on profile. Form-based single sign-on allows users to log on one time to all protected applications in your network, instead of requiring them to log on separately to access each one.

HDX

Whether or not HDX Proxy for ICA traffic is enabled.

fta

Whether or not file-type association is enabled.

wanscaler

Use the Repeater Plug-in to optimize network traffic.

kcdAccount

Kerberos constrained delegation account name

samlSSOProfile

Profile to be used for doing SAML SSO to remote relying party

proxy

IP address and Port of the proxy server to be used for HTTP access for this request.

userExpression

expression that will be evaluated to obtain username for SingleSignOn

passwdExpression

expression that will be evaluated to obtain password for SingleSignOn

stateflag

devno

count

vpn trafficPolicy

The following operations can be performed on "vpn trafficPolicy":

[add](#) | [rm](#) | [set](#) | [unset](#) | [show](#)

add vpn trafficPolicy

Creates a traffic policy. A traffic policy conditionally sets NetScaler Gateway traffic characteristics at run time. For an intranet resource, for example, the traffic policy parameters define the destination IP address, destination port, amount of time a user can stay logged on to the application, and HTTP compression.

Synopsys

add vpn trafficPolicy <name> <rule> <action>

Arguments

name

Name for the traffic policy. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Cannot be changed after the policy is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my policy" or 'my policy').

rule

Expression, or name of a named expression, against which traffic is evaluated. Written in the classic or default syntax.

Note:

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

The following requirements apply only to the NetScaler CLI:

- * If the expression includes one or more spaces, enclose the entire expression in double quotation marks.
- * If the expression itself includes double quotation marks, escape the quotations by using the \ character.
- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Action to apply to traffic that matches the policy.

rm vpn trafficPolicy

Removes an existing traffic policy from NetScaler Gateway.

Synopsys

rm vpn trafficPolicy <name>

Arguments

name

Name of the traffic policy to remove.

set vpn trafficPolicy

Modifies the specified parameters of an existing traffic policy.

Synopsis

```
set vpn trafficPolicy <name> [-rule <expression>] [-action <string>]
```

Arguments

name

Name of the traffic policy to modify.

rule

Expression, or name of a named expression, against which traffic is evaluated. Written in the classic or default syntax.

Note:

Maximum length of a string literal in the expression is 255 characters. A longer string can be split into smaller strings of up to 255 characters each, and the smaller strings concatenated with the + operator. For example, you can create a 500-character string as follows: "<string of 255 characters>" + "<string of 245 characters>"

The following requirements apply only to the NetScaler CLI:

- * If the expression includes one or more spaces, enclose the entire expression in double quotation marks.
- * If the expression itself includes double quotation marks, escape the quotations by using the \ character.
- * Alternatively, you can use single quotation marks to enclose the rule, in which case you do not have to escape the double quotation marks.

action

Action to apply to traffic that matches the policy.

unset vpn trafficPolicy

Use this command to remove vpn trafficPolicy settings. Refer to the set vpn trafficPolicy command for meanings of the arguments.

Synopsis

```
unset vpn trafficPolicy <name> [-rule] [-action]
```

show vpn trafficPolicy

Displays information about all NetScaler Gateway traffic policies, or detailed information about the specified policy.

Synopsis

```
show vpn trafficPolicy [<name>]
```

Arguments

name

Name of the traffic policy for which to display detailed information.

Outputs

rule

The rule used by the vpn traffic policy. Rules are combinations of expressions. Expressions are simple conditions, such as a test for equality, applied to operands, such as a URL string or an IP address. Expression syntax is described in the Installation and Configuration Guide

action

The action to be performed when the rule is matched.

boundTo

The entity name to which policy is bound

activePolicy

priority

bindPolicyType

policyType

devno

count

stateflag

vpn url

The following operations can be performed on "vpn url":

add | **rm** | **set** | **unset** | **show**

add vpn url

Creates a bookmark link to an external or internal resource that appears on the Access Interface, according to type, as a web site link or file share link.

Synopsys

```
add vpn url <urlName> <linkName> <actualURL> [-clientlessAccess ( ON | OFF )] [-comment <string>]
```

Arguments

urlName

Name of the bookmark link.

linkName

Description of the bookmark link. The description appears in the Access Interface.

actualURL

Web address for the bookmark link.

clientlessAccess

If clientless access to the resource hosting the link is allowed, also use clientless access for the bookmarked web address in the Secure Client Access based session. Allows single sign-on and other HTTP processing on NetScaler Gateway for HTTPS resources.

Possible values: ON, OFF

Default value: OFF

comment

Any comments associated with the bookmark link.

Example

```
add vpn url ggl search www.google.com.
```

rm vpn url

Removes a bookmark link to an internal resource that appears in the Access Interface.

Synopsys

```
rm vpn url <urlName>
```

Arguments

urlName

Name of the bookmark link to remove.

Example

```
rm vpn url ggl
```

set vpn url

Modifies the specified parameters of a bookmark link to an internal resource that appears in the Access Interface.

Synopsis

```
set vpn url <urlName> [-linkName <string>] [-actualURL <string>] [-clientlessAccess ( ON | OFF )] [-comment <string>]
```

Arguments

urlName

Name of the bookmark link.

linkName

Description of the bookmark link. The description appears in the Access Interface.

actualURL

Web address for the bookmark link.

clientlessAccess

If clientless access to the resource hosting the link is allowed, also use clientless access for the bookmarked web address in the Secure Client Access based session. Allows single sign-on and other HTTP processing on NetScaler Gateway for HTTPS resources.

Possible values: ON, OFF

Default value: OFF

comment

Any comments associated with the bookmark link.

Example

```
set vpn url wiurl -clientlessAccess on
```

unset vpn url

Use this command to remove vpn url settings. Refer to the set vpn url command for meanings of the arguments.

Synopsis

```
unset vpn url <urlName> [-clientlessAccess] [-comment]
```

show vpn url

Displays information about all the configured bookmark links to internal resources that appear in the Access Interface, or displays detailed information about the specified bookmark link.

Synopsis

```
show vpn url [<urlName>]
```

Arguments

urlName

Name of the bookmark link for which to display detailed information.

Outputs

linkName

Description of the bookmark link. The description appears in the Access Interface.

actualURL

Web address for the bookmark link.

clientlessAccess

Whether clientless access is enabled for the url in other modes or not.

comment

Comments associated with this virtual server.

devno

count

stateflag

vpn vserver

The following operations can be performed on "vpn vserver":

add | **rm** | **set** | **unset** | **bind** | **unbind** | **enable** | **disable** | **show** | **stat** | **rename** | **check**

add vpn vserver

Creates a NetScaler Gateway virtual server to allow authenticated users to access intranet resources, such as XenApp, XenDesktop, and web servers.

Synopsys

```
add vpn vserver <name> <serviceType> (<IPAddress> [-range <positive_integer>]) <port> [-state ( ENABLED |
DISABLED )] [-authentication ( ON | OFF )] [-doubleHop ( ENABLED | DISABLED )] [-maxAAUsers
<positive_integer>] [-icaOnly ( ON | OFF )] [-icaProxySessionMigration ( ON | OFF )] [-deviceCert ( ON | OFF )] [-
certkeyNames <string>] [-downStateFlush ( ENABLED | DISABLED )] [-Listenpolicy <expression>] [-Listenpriority
<positive_integer>] [-tcpProfileName <string>] [-httpProfileName <string>] [-comment <string>] [-appflowLog (
ENABLED | DISABLED )] [-icmpVsrResponse ( PASSIVE | ACTIVE )] [-RHlstate ( PASSIVE | ACTIVE )] [-netProfile
<string>] [-cginfraHomePageRedirect ( ENABLED | DISABLED )] [-maxLoginAttempts <positive_integer>] [-
failedLoginTimeout <mins>] [-l2Conn ( ON | OFF )] [-deploymentType <deploymentType>]
```

Arguments

name

Name for the NetScaler Gateway virtual server. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters. Can be changed after the virtual server is created.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my server" or 'my server').

serviceType

Protocol used by the NetScaler Gateway virtual server.

Possible values: SSL

Default value: SSL

IPAddress

IPv4 or IPv6 address of the NetScaler Gateway virtual server. Usually a public IP address. User devices send connection requests to this IP address.

range

Range of NetScaler Gateway virtual server IP addresses. The consecutively numbered range of IP addresses begins with the address specified by the IP Address parameter.

In the configuration utility, select Network VServer to enter a range.

Default value: 1

Minimum value: 1

port

TCP port on which the virtual server listens.

Minimum value: 1

state

State of the virtual server. If the virtual server is disabled, requests are not processed.

Possible values: ENABLED, DISABLED

Default value: ENABLED

authentication

Require authentication for users connecting to NetScaler Gateway.

Possible values: ON, OFF

Default value: ON

doubleHop

Use the NetScaler Gateway appliance in a double-hop configuration. A double-hop deployment provides an extra layer of security for the internal network by using three firewalls to divide the DMZ into two stages. Such a deployment can have one appliance in the DMZ and one appliance in the secure network.

Possible values: ENABLED, DISABLED

Default value: DISABLED

maxAAUsers

Maximum number of concurrent user sessions allowed on this virtual server. The actual number of users allowed to log on to this virtual server depends on the total number of user licenses.

Minimum value: 0

icaOnly

User can log on in Basic mode only, through either Citrix Receiver or a browser. Users are not allowed to connect by using the NetScaler Gateway Plug-in.

Possible values: ON, OFF

Default value: OFF

icaProxySessionMigration

This option determines if an existing ICA Proxy session is transferred when the user logs on from another device.

Possible values: ON, OFF

Default value: OFF

deviceCert

Indicates whether device certificate check as a part of EPA is on or off.

Possible values: ON, OFF

Default value: OFF

certkeyNames

Name of the certificate key that was bound to the corresponding SSL virtual server as the Certificate Authority for the device certificate

downStateFlush

Close existing connections when the virtual server is marked DOWN, which means the server might have timed out. Disconnecting existing connections frees resources and in certain cases speeds recovery of overloaded load balancing setups. Enable this setting on servers in which the connections can safely be closed when they are marked DOWN. Do not enable DOWN state flush on servers that must complete their transactions.

Possible values: ENABLED, DISABLED

Default value: ENABLED

Listenpolicy

String specifying the listen policy for the NetScaler Gateway virtual server. Can be either a named expression or a default syntax expression. The NetScaler Gateway virtual server processes only the traffic for which the expression evaluates to true.

Default value: "none"

Listenpriority

Integer specifying the priority of the listen policy. A higher number specifies a lower priority. If a request matches the listen policies of more than one virtual server, the virtual server whose listen policy has the highest priority (the lowest priority number) accepts the request.

Default value: 101

Minimum value: 0

Maximum value: 100

tcpProfileName

Name of the TCP profile to assign to this virtual server.

httpProfileName

Name of the HTTP profile to assign to this virtual server.

comment

Any comments associated with the virtual server.

appflowLog

Log AppFlow records that contain standard NetFlow or IPFIX information, such as time stamps for the beginning and end of a flow, packet count, and byte count. Also log records that contain application-level information, such as HTTP web addresses, HTTP request methods and response status codes, server response time, and latency.

Possible values: ENABLED, DISABLED

Default value: DISABLED

icmpVsrResponse

Criterion for responding to PING requests sent to this virtual server. If this parameter is set to ACTIVE, respond only if the virtual server is available. With the PASSIVE setting, respond even if the virtual server is not available.

Possible values: PASSIVE, ACTIVE

Default value: PASSIVE

RHlstate

A host route is injected according to the setting on the virtual servers.

* If set to PASSIVE on all the virtual servers that share the IP address, the appliance always injects the hostroute.

* If set to ACTIVE on all the virtual servers that share the IP address, the appliance injects even if one virtual server is UP.

* If set to ACTIVE on some virtual servers and PASSIVE on the others, the appliance injects even if one virtual server set to ACTIVE is UP.

Possible values: PASSIVE, ACTIVE

Default value: PASSIVE

netProfile

The name of the network profile.

cginfraHomePageRedirect

When client requests ShareFile resources and NetScaler Gateway detects that the user is unauthenticated or the user session has expired, disabling this option takes the user to the originally requested ShareFile resource after authentication (instead of taking the user to the default VPN home page)

Possible values: ENABLED, DISABLED

Default value: ENABLED

maxLoginAttempts

Maximum number of logon attempts

Minimum value: 1

Maximum value: 255

failedLoginTimeout

Number of minutes an account will be locked if user exceeds maximum permissible attempts

Minimum value: 1

l2Conn

Use Layer 2 parameters (channel number, MAC address, and VLAN ID) in addition to the 4-tuple (<source IP>:<source port>:<destination IP>:<destination port>) that is used to identify a connection. Allows multiple TCP and non-TCP connections with the same 4-tuple to coexist on the NetScaler appliance.

Possible values: ON, OFF

deploymentType

Example

The following example creates a VPN virtual server named myvpnvip which supports SSL prot

rm vpn vserver

Removes a NetScaler Gateway virtual server. Policies that are bound to the virtual server are automatically unbound.

Synopsys

```
rm vpn vserver <name>@ ...
```

Arguments

name

Name of the virtual server to remove.

Example

```
rm vserver vpn_vip
```

set vpn vserver

Modifies the specified parameters of a NetScaler Gateway virtual server.

Synopsys


```
set vpn vserver <name> [-IPAddress <ip_addr|ipv6_addr|*>] [-authentication ( ON | OFF )] [-doubleHop ( ENABLED | DISABLED )] [-icaOnly ( ON | OFF )] [-icaProxySessionMigration ( ON | OFF )] [-deviceCert ( ON | OFF )] [-certkeyNames <string>] [-maxAAAUsers <positive_integer>] [-downStateFlush ( ENABLED | DISABLED )] [-Listenpolicy <expression>] [-Listenpriority <positive_integer>] [-tcpProfileName <string>] [-httpProfileName <string>] [-comment <string>] [-appflowLog ( ENABLED | DISABLED )] [-icmpVsrResponse ( PASSIVE | ACTIVE )] [-RHlstate ( PASSIVE | ACTIVE )] [-netProfile <string>] [-cginfraHomePageRedirect ( ENABLED | DISABLED )] [-maxLoginAttempts <positive_integer>] [-failedLoginTimeout <mins>] [-l2Conn ( ON | OFF )]
```

Arguments

name

Name of the virtual server to modify.

IPAddress

IPv4 or IPv6 address of the NetScaler Gateway virtual server. Usually a public IP address. User devices send connection requests to this IP address.

authentication

Require authentication for users connecting to NetScaler Gateway.

Possible values: ON, OFF

Default value: ON

doubleHop

Use the NetScaler Gateway appliance in a double-hop configuration. A double-hop deployment provides an extra layer of security for the internal network by using three firewalls to divide the DMZ into two stages. Such a deployment can have one appliance in the DMZ and one appliance in the secure network.

Possible values: ENABLED, DISABLED

Default value: DISABLED

icaOnly

User can log on in Basic mode only, through either Citrix Receiver or a browser. Users are not allowed to connect by using the NetScaler Gateway Plug-in.

Possible values: ON, OFF

Default value: OFF

icaProxySessionMigration

This option determines if an existing ICA Proxy session is transferred when the user logs on from another device.

Possible values: ON, OFF

Default value: OFF

deviceCert

Indicates whether device certificate check as a part of EPA is enabled or not.

Possible values: ON, OFF

Default value: OFF

certkeyNames

Name of the certkey which was bound to the corresponding SSL virtual server as the Certificate Authority for the device certificate

maxAAAUsers

Maximum number of concurrent user sessions allowed on this virtual server. The actual number of users allowed to log on to this virtual server depends on the total number of user licenses.

Minimum value: 0

downStateFlush

Close existing connections when the virtual server is marked DOWN, which means the server might have timed out. Disconnecting existing connections frees resources and in certain cases speeds recovery of overloaded load balancing setups. Enable this setting on servers in which the connections can safely be closed when they are marked DOWN. Do not enable DOWN state flush on servers that must complete their transactions.

Possible values: ENABLED, DISABLED

Default value: ENABLED

Listenpolicy

String specifying the listen policy for the NetScaler Gateway virtual server. Can be either a named expression or a default syntax expression. The NetScaler Gateway virtual server processes only the traffic for which the expression evaluates to true.

Default value: "none"

Listenpriority

Integer specifying the priority of the listen policy. A higher number specifies a lower priority. If a request matches the listen policies of more than one virtual server, the virtual server whose listen policy has the highest priority (the lowest priority number) accepts the request.

Default value: 101

Minimum value: 0

Maximum value: 100

tcpProfileName

Name of the TCP profile to assign to this virtual server.

httpProfileName

Name of the HTTP profile to assign to this virtual server.

comment

Any comments associated with the virtual server.

appflowLog

Log AppFlow records that contain standard NetFlow or IPFIX information, such as time stamps for the beginning and end of a flow, packet count, and byte count. Also log records that contain application-level information, such as HTTP web addresses, HTTP request methods and response status codes, server response time, and latency.

Possible values: ENABLED, DISABLED

Default value: DISABLED

icmpVsrResponse

Criterion for responding to PING requests sent to this virtual server. If this parameter is set to ACTIVE, respond only if the virtual server is available. With the PASSIVE setting, respond even if the virtual server is not available.

Possible values: PASSIVE, ACTIVE

Default value: PASSIVE

RHIstate

A host route is injected according to the setting on the virtual servers.

* If set to PASSIVE on all the virtual servers that share the IP address, the appliance always injects the hostroute.

* If set to ACTIVE on all the virtual servers that share the IP address, the appliance injects even if one virtual server is UP.

* If set to ACTIVE on some virtual servers and PASSIVE on the others, the appliance injects even if one virtual server set to ACTIVE is UP.

Possible values: PASSIVE, ACTIVE

Default value: PASSIVE

netProfile

The name of the network profile.

cginfraHomePageRedirect

When client requests ShareFile resources and NetScaler Gateway detects that the user is unauthenticated or the user session has expired, disabling this option takes the user to the originally requested ShareFile resource after authentication (instead of taking the user to the default VPN home page)

Possible values: ENABLED, DISABLED

Default value: ENABLED

maxLoginAttempts

Maximum number of logon attempts

Minimum value: 1

Maximum value: 255

failedLoginTimeout

Number of minutes an account will be locked if user exceeds maximum permissible attempts

Minimum value: 1

I2Conn

Use Layer 2 parameters (channel number, MAC address, and VLAN ID) in addition to the 4-tuple (<source IP>:<source port>:<destination IP>:<destination port>) that is used to identify a connection. Allows multiple TCP and non-TCP connections with the same 4-tuple to coexist on the NetScaler appliance.

Possible values: ON, OFF

unset vpn vserver

Use this command to remove vpn vserver settings. Refer to the set vpn vserver command for meanings of the arguments.

Synopsys

```
unset vpn vserver <name> [-authentication] [-doubleHop] [-icaOnly] [-icaProxySessionMigration] [-deviceCert] [-certkeyNames] [-maxAAAUsers] [-downStateFlush] [-Listenpolicy] [-Listenpriority] [-tcpProfileName] [-httpProfileName] [-comment] [-appflowLog] [-icmpVsrResponse] [-RHIstate] [-netProfile] [-cginfraHomePageRedirect] [-maxLoginAttempts] [-I2Conn]
```

bind vpn vserver

Binds attributes to the specified NetScaler Gateway virtual server.

Synopsys

```
bind vpn vserver <name> [-policy <string> [-priority <positive_integer>] [-secondary] [-groupExtraction] [-gotoPriorityExpression <expression>] [-type <type>]] [-intranetApplication <string>] [-nextHopServer <string>] [-urlName <string>] [-intranetIP <ip_addr> <netmask> ] [-staServer <URL> [-staAddressType ( IPV4 | IPV6 )]] [-appController <URL>] [-sharefile <string>]
```

Arguments

name

Name of the virtual server.

policy

Name of a policy to bind to the virtual server (for example, the name of an authentication, session, or endpoint analysis policy).

priority

Integer specifying the policy's priority. The lower the number, the higher the priority. Policies are evaluated in the order of their priority numbers.

Minimum value: 0

secondary

Binds the authentication policy as the secondary policy to use in a two-factor configuration. A user must then authenticate not only via a primary authentication method but also via a secondary authentication method. User groups are aggregated across both. The user name must be exactly the same for both authentication methods, but they can require different passwords.

groupExtraction

Binds the authentication policy to a tertiary chain which will be used only for group extraction. The user will not authenticate against this server, and this will only be called if primary and/or secondary authentication has succeeded.

gotoPriorityExpression

Expression or other value specifying the next policy to evaluate if the current policy evaluates to TRUE. Specify one of the following values:

- * NEXT - Evaluate the policy with the next higher priority number.
- * END - End policy evaluation.
- * USE_INVOCATION_RESULT - Applicable if this policy invokes another policy label. If the final goto in the invoked policy label has a value of END, the evaluation stops. If the final goto is anything other than END, the current policy label performs a NEXT.
- * A default syntax or classic expression that evaluates to a number.

If you specify an expression, the number to which it evaluates determines the next policy to evaluate, as follows:

- * If the expression evaluates to a higher numbered priority, the policy with that priority is evaluated next.
- * If the expression evaluates to the priority of the current policy, the policy with the next higher numbered priority is evaluated next.
- * If the expression evaluates to a number that is larger than the largest numbered priority, policy evaluation ends.

An UNDEF event is triggered if:

- * The expression is invalid.
- * The expression evaluates to a priority number that is numerically lower than the current policy's priority.

* The expression evaluates to a priority number that is between the current policy's priority number (say, 30) and the highest priority number (say, 100), but does not match any configured priority number (for example, the expression evaluates to the number 85). This example assumes that the priority number increments by 10 for every successive policy, and therefore a priority number of 85 does not exist in the policy label.

type

Bind point to which to bind the policy. Applies only to rewrite and cache policies. If you do not set this parameter, the policy is bound to REQ_DEFAULT or RES_DEFAULT, depending on whether the policy rule is a response-time or a request-time expression.

Possible values: REQUEST, RESPONSE, ICA_REQUEST, OTHERTCP_REQUEST

intranetApplication

Name of the application to bind to the virtual server. Intranet applications are used to enable access to selected applications located in the internal network. They are required for any user connecting with the NetScaler Gateway Plug-in for Java.

nextHopServer

Name of the next hop server to bind to the virtual server.

urlName

Web address of the next hop virtual server to bind to the virtual server.

intranetIP

The network ID for the range of intranet IP addresses or individual intranet IP addresses to be bound to the virtual server.

netmask

A range of IP addresses in an address pool, bound to a virtual server. When users log on, NetScaler Gateway assigns an IP address from the pool.

staServer

Web address of the Secure Ticket Authority (STA) server, in the following format: 'http(s)://FQDN/URLPATH'

staAddressType

Type of the STA server address(ipv4/v6).

Possible values: IPV4, IPV6

appController

App Controller server, in the format 'http(s)://IP/FQDN'

sharefile

ShareFile server, in the format 'IP:PORT / FQDN:PORT'

unbind vpn vserver

Unbinds the specified attributes from a virtual server.

Synopsys

```
unbind vpn vserver <name> [-policy <string> [-secondary] [-groupExtraction] [-type <type>]] [-intranetApplication <string>] [-nextHopServer <string>] [-urlName <string>] [-intranetIP <ip_addr> <netmask>] [-staServer <URL>] [-appController <URL>] [-sharefile <string>]
```

Arguments

name

Name of the virtual server from which to unbind an attribute.

policy

Name of the policy to unbind from the virtual server.

secondary

Binds the authentication policy as the secondary policy to use in a two-factor configuration. A user must then authenticate not only via a primary authentication method but also via a secondary authentication method. User groups are aggregated across both. The user name must be exactly the same for both authentication methods, but they can require different passwords.

groupExtraction

Binds the authentication policy to a tertiary chain which will be used only for group extraction. The user will not authenticate against this server, and this will only be called if primary and/or secondary authentication has succeeded.

type

Bind point from which to unbind the policy.

Possible values: REQUEST, RESPONSE, ICA_REQUEST, OTHERTCP_REQUEST

intranetApplication

Name of intranet application to unbind from the virtual server.

nextHopServer

Name of the next hop server to remove.

urlName

Web address of the next hop virtual server to unbind.

intranetIP

The range of IP addresses to unbind from the virtual server.

netmask

The netmask of the intranet IP address or range.

staServer

Web address of the Secure Ticket Authority (STA) server to remove, in the following format: 'http(s)://FQDN/URLPATH'

appController

App Controller server to be removed, in the format 'http(s)://IP/FQDN'

sharefile

ShareFile server to be removed, in the format 'IP:PORT / FQDN:PORT'

enable vpn vserver

Enables a NetScaler Gateway virtual server. Note: Virtual servers, when added, are enabled by default.

Synopsys

enable vpn vserver <name>@

Arguments

name

Name of the virtual server to be enabled.

Example

```
enable vserver vpn1
```

disable vpn vserver

Disables a NetScaler Gateway virtual server. The virtual server is taken out of service.

Synopsys

```
disable vpn vserver <name>@
```

Arguments

name

Name of the virtual server to be disabled. The NetScaler Gateway still responds to ARP and/or PING requests for the IP address of the virtual server. You can enable the NetScaler Gateway virtual server again at any time, because the virtual server is still configured.

Example

```
disable vserver lb_vip
```

show vpn vserver

Displays information about all the configured NetScaler Gateway virtual servers, or displays detailed information about the specified NetScaler Gateway virtual server.

Synopsys

```
show vpn vserver [<name>] show vpn vserver stats - alias for 'stat vpn vserver'
```

Arguments

name

Name of the NetScaler Gateway virtual server for which to show detailed information.

Outputs

IPAddress

The IP address of the virtual server.

value

Indicates whether or not the certificate is bound or if SSL offload is disabled.

port

The virtual TCP port of the VPN virtual server.

range

The range of VPN virtual server IP addresses. The new range of VPN virtual servers will have IP addresses consecutively numbered, starting with the primary address specified with the <ipaddress> argument.

serviceType

The VPN virtual server's protocol type. Currently, the only possible value is SSL.

type

Bindpoint to which the policy is bound.

state

State of the virtual server. If the virtual server is disabled, requests are not processed.

status

Whether or not this virtual server responds to ARPs and whether or not round-robin selection is temporarily in effect.

cacheType

Virtual server cache type. The options are: TRANSPARENT, REVERSE, and FORWARD.

redirect

The cache redirect policy.

The valid redirect policies are:

1. CACHE - Directs all requests to the cache.
2. POLICY - Applies cache redirection policy to determine whether the request should be directed to the cache or origin. This is the default setting.
3. ORIGIN - Directs all requests to the origin server.

precedence

This argument is used only when configuring content switching on the specified virtual server. This is applicable only

if both the URL and RULE-based policies have been configured on the same virtual server.

It specifies the type of policy (URL or RULE) that takes precedence on the content switching virtual server. The default setting is RULE.

I URL - In this case, the incoming request is matched against the URL-based policies before the rule-based policies.

I RULE - In this case, the incoming request is matched against the rule-based policies before the URL-based policies.

For all URL-based policies, the precedence hierarchy is:

1. Domain and exact URL
2. Domain, prefix, and suffix
3. Domain and suffix
4. Domain and prefix
5. Domain only
6. Exact URL
7. Prefix and suffix
8. Suffix only
9. Prefix only
10. Default

redirectURL

The URL where traffic is redirected if the virtual server in system becomes unavailable. WARNING! Make sure that the domain you specify in the URL does not match the domain specified in the -d domainName argument of the ###add cs policy### command. If the same domain is specified in both arguments, the request will be continuously redirected to the same unavailable virtual server in the system. If so, the user may not get the requested content.

authentication

Indicates whether or not authentication is being applied to incoming users to the VPN.

doubleHop

Indicates whether double-hop functionality is enabled or not.

icaOnly

Indicates whether an ICA only license feature is enabled or not.

icaProxySessionMigration

This option determines if an existing ICA Proxy session is transferred when the user logs on from another device.

advancedEpa

Indicates whether advanced EPA feature is enabled or not.

deviceCert

Indicates whether device certificate check as a part of EPA is enabled or not.

certkeyNames

Name of the certificate key which was bound to the corresponding SSL virtual server as the Certificate Authority for the device certificate

maxAAAUUsers

The maximum number of concurrent users allowed to log on into this virtual server at a time.

curAAAUUsers

The number of current users logged on to this virtual server.

curTotalUsers

The total number of current users connected through this virtual server.

domain

The domain name of the server for which a service needs to be added. If the IP address has been specified, the domain name does not need to be specified.

rule

The name of the rule, or expression, if any, that policy for the VPN server is to use. Rules are combinations of expressions. Expressions are simple conditions, such as a test for equality, applied to operands, such as a URL string or an IP address. Expression syntax is described in the Installation and Configuration Guide. The default rule is ns_true.

policyName

The name of the policy, if any, bound to the VPN virtual server.

policy

The name of the policy, if any, bound to the VPN virtual server.

serviceName

The name of the service, if any, to which the virtual server policy is bound.

weight

Weight for this service, if any. This weight is used when the system performs load balancing, giving greater priority to a specific service. It is useful when the services bound to a virtual server are of different capacity.

cacheVserver

The name of the default target cache virtual server, if any, to which requests are redirected.

backupVServer

The name of the backup VPN virtual server for this VPN virtual server.

priority

The priority, if any, of the VPN virtual server policy.

cltTimeout

The idle time, if any, in seconds after which the client connection is terminated.

soMethod

VPN client applications are allocated from a block of intranet IP addresses.

That block may be exhausted after a certain number of connections. This switch specifies the method used to determine whether or not a new connection will spill over, or exhaust, the allocated block of intranet IP addresses for that application. Possible values are CONNECTION or DYNAMICCONNECTION. CONNECTION means that a static integer value is the hard limit for the spillover threshold. The spillover threshold is described below. DYNAMICCONNECTION means that the spillover threshold is set according to the maximum number of connections defined for the VPN virtual server.

soThreshold

VPN client applications are allocated from a block of intranet IP addresses.

That block may be exhausted after a certain number of connections.

The value of this option is the number of client connections after which the mapped IP address is used as the client source IP address instead of an address from the allocated block of intranet IP addresses.

soPersistence

Whether or not cookie-based site persistence is enabled for this VPN vserver. Possible values are 'ConnectionProxy', HTTPRedirect, or NONE

soPersistenceTimeOut

The timeout, if any, for cookie-based site persistence of this VPN vserver.

actType**intranetApplication**

The intranet VPN application.

nextHopServer

The name of the next hop server bound to the VPN virtual server.

urlName

The intranet URL.

intranetIP

The network ID for the range of intranet IP addresses or individual intranet IP addresses to be bound to the virtual server.

netmask

The netmask of the intranet IP address or range.

staServer

Configured Secure Ticketing Authority (STA) server.

staAddressType

Type of the STA server address(ipv4/v6).

staAuthID

Authority ID of the STA Server. Authority ID is used to match incoming STA tickets in the SOCKS/CGP protocol with the right STA server.

appController

Configured App Controller server in XenMobile deployment.

sharefile

Configured ShareFile server in XenMobile deployment. Format IP:PORT / FQDN:PORT

useMIP

Deprecated. See 'map' below.

map

Whether or not mapped IP addresses are ON or OFF. Mapped IP addresses are source IP addresses

for the virtual servers running on the NetScaler. Mapped IP addresses are used by the system to connect to the backend servers.

downStateFlush

Close existing connections when the virtual server is marked DOWN, which means the server might have timed out. Disconnecting existing connections frees resources and in certain cases speeds recovery of overloaded load balancing setups. Enable this setting on servers in which the connections can safely be closed when they are marked DOWN. Do not enable DOWN state flush on servers that must complete their transactions.

gotoPriorityExpression

Next priority expression.

disablePrimaryOnDown

Tells whether traffic will continue reaching backup virtual servers even after the primary virtual server comes UP from DOWN state.

Listenpolicy

The string is listenpolicy configured for VPN vserver

Listenpriority

This parameter is the priority for listen policy of VPN Vserver.

tcpProfileName

Name of the TCP profile to assign to this virtual server.

httpProfileName

Name of the HTTP profile to assign to this virtual server.

policySubType

stateflag

flags

comment

Any comments associated with the virtual server.

appflowLog

Log AppFlow records that contain standard NetFlow or IPFIX information, such as time stamps for the beginning and end of a flow, packet count, and byte count. Also log records that contain application-level information, such as HTTP web addresses, HTTP request methods and response status codes, server response time, and latency.

icmpVsrResponse

Criterion for responding to PING requests sent to this virtual server. If this parameter is set to ACTIVE, respond only if the virtual server is available. With the PASSIVE setting, respond even if the virtual server is not available.

RHlstate

A host route is injected according to the setting on the virtual servers.

* If set to PASSIVE on all the virtual servers that share the IP address, the appliance always injects the hostroute.

* If set to ACTIVE on all the virtual servers that share the IP address, the appliance injects even if one virtual server is UP.

* If set to ACTIVE on some virtual servers and PASSIVE on the others, the appliance injects even if one virtual server set to ACTIVE is UP.

netProfile

The name of the network profile.

cginfraHomePageRedirect

When client requests ShareFile resources and NetScaler Gateway detects that the user is unauthenticated or the user session has expired, disabling this option takes the user to the originally requested ShareFile resource after authentication (instead of taking the user to the default VPN home page)

maxLoginAttempts

Maximum number of logon attempts

failedLoginTimeout

Number of minutes an account will be locked if user exceeds maximum permissible attempts

secondary

Binds the authentication policy as the secondary policy to use in a two-factor configuration. A user must then authenticate not only via a primary authentication method but also via a secondary authentication method. User groups are aggregated across both. The user name must be exactly the same for both authentication methods, but they can require different passwords.

groupExtraction

Binds the authentication policy to a tertiary chain which will be used only for group extraction. The user will not authenticate against this server, and this will only be called if primary and/or secondary authentication has succeeded.

deploymentType

epaprofile

Advanced EPA profile to bind

epaprofileoptional

Mark the EPA profile optional for preauthentication EPA profile. User would be shown a logon page even if the EPA profile fails to evaluate.

ngname

Node group devno to which this authentication virtual sever belongs

vstype

Virtual Server Type, such as Load Balancing, Content Switch, Cache Redirection

l2Conn

Use Layer 2 parameters (channel number, MAC address, and VLAN ID) in addition to the 4-tuple (<source IP>:<source port>:<destination IP>:<destination port>) that is used to identify a connection. Allows multiple TCP and non-TCP connections with the same 4-tuple to coexist on the NetScaler appliance.

devno**count**

Example

```
show vpn vserver
```

stat vpn vserver

Displays statistics for all NetScaler Gateway virtual servers, or displays detailed statistics for the specified NetScaler Gateway virtual server.

Synopsys

```
stat vpn vserver [<name>] [-detail] [-fullValues] [-ntimes <positive_integer>] [-logFile <input_filename>] [-clearstats (basic | full )]
```

Arguments

name

Name of the virtual server for which to show detailed statistics.

detail

Specifies detailed output (including more statistics). The output can be quite voluminous. Without this argument, the output will show only a summary.

fullValues

Specifies that numbers and strings should be displayed in their full form. Without this option, long strings are shortened and large numbers are abbreviated

ntimes

The number of times, in intervals of seven seconds, the statistics should be displayed.

Default value: 1

Minimum value: 0

logFile

The name of the log file to be used as input.

clearstats

Clear the statistics / counters

Possible values: basic, full

Outputs

count

devno

stateflag

Outputs

IP address (IP)

The IP address on which the service is running.

Port (port)

The port on which the service is running.

Vserver protocol (Protocol)

Protocol associated with the vservers

State

Current state of the server. Possible values are UP, DOWN, UNKNOWN, OFS(Out of Service), TROFS (Transition Out of Service), TROFS_DOWN(Down When going Out of Service)

Requests (Req)

Total number of requests received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

Responses (Rsp)

Number of responses received on this service or virtual server. (This applies to HTTP/SSL services and servers.)

Request bytes (Reqb)

Total number of request bytes received on this service or virtual server.

Response bytes (Rspb)

Number of response bytes received by this service or virtual server.

rename vpn vservers

Renames a NetScaler Gateway virtual server.

Synopsis

```
rename vpn vservers <name>@ <newName>@
```

Arguments

name

Name of the NetScaler Gateway virtual server.

newName

New name for the NetScaler Gateway virtual server. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

The following requirement applies only to the NetScaler CLI:

If the name includes one or more spaces, enclose the name in double or single quotation marks (for example, "my server" or 'my server').

Example

```
rename vpn vserver vpn1 vpn1new
```

check vpn vserver

Invokes Cerebro executable for connectivity checks for the servers bound to a VPN virtual server

Synopsys

```
check vpn vserver <name>
```

Arguments

name

Name of the NetScaler Gateway virtual server.

Outputs

response

Example

```
check vpn vserver <vserver name>
```

WebInterface Commands

The entities on which you can perform NetScaler CLI operations:

- [wi package](#)
- [wi site](#)

wi package

The following operations can be performed on "wi package":

install | **uninstall**

install wi package

Installs Web Interface and JRE tar files on the NetScaler appliance.

Synopsys

```
install wi package [-jre <URL>] [-wi <URL>] [-maxSites <maxSites>]
```

Arguments

jre

Complete path to the JRE tar file.

You can use the Diablo Latte JRE version 1.6.0-7 for 64-bit FreeBSD 6.x/amd64 platform available on the FreeBSD Foundation web site.

Alternatively, you can use OpenJDK6 package for FreeBSD 6.x/amd63. The Java package can be downloaded from http://ftp.riken.jp/pub/FreeBSD/ports/amd64/packages-6-stable/java/openjdk6-b17_2.tbz or <http://www.freebsdoundation.org/cgi-bin/download?download=diablo-jdk-freebsd6.amd64.1.6.0.07.02.tbz>

Default value: "file://tmp/diablo-jdk-freebsd6.amd64.1.6.0.07.02.tbz"

wi

Complete path to the Web Interface tar file for installing the Web Interface on the NetScaler appliance. This file includes Apache Tomcat Web server. The file name has the following format: nswi-<version number>.tgz (for example, nswi-1.5.tgz).

Default value: "http://citrix.com/downloads/nswi-1.7.tgz"

maxSites

Maximum number of Web Interface sites that can be created on the NetScaler appliance; changes the amount of RAM reserved for Web Interface usage; changing its value results in restart of Tomcat server and invalidates any existing Web Interface sessions.

Possible values: 3, 25, 50, 100, 200, 500

Example

```
install wi package -jre http://10.102.1.10/diablo-latte-freebsd6-amd64-1.6.0_07-b02.tar.b:
```

uninstall wi package

Removes the Web Interface and JRE tar files, and the entire Web Interface related configuration, from the NetScaler appliance.

Synopsys

```
uninstall wi package
```

Example

```
uninstall wi package
```

wi site

The following operations can be performed on "wi site":

add | **rm** | **set** | **unset** | **bind** | **unbind** | **show**

add wi site

Creates a Web Interface site on the NetScaler appliance. The NetScaler Web Interface feature provides access to Citrix XenApp and Citrix XenDesktop applications. Users access resources through a standard web browser or by using the Citrix XenApp plug-in.

Synopsys

```
add wi site <sitePath> [<agURL> [<staURL> [-secondSTAURL <string> [-useTwoTickets ( ON | OFF )]] [-sessionReliability ( ON | OFF )]] [-authenticationPoint ( WebInterface | AccessGateway ) [-agAuthenticationMethod ( Explicit | SmartCard )]] [-wiAuthenticationMethods ( Explicit | Anonymous ) ...] [-defaultCustomTextLocale <defaultCustomTextLocale>] [-webSessionTimeout <positive_integer>] [-defaultAccessMethod <defaultAccessMethod>] [-loginTitle <string>] [-appWelcomeMessage <string>] [-welcomeMessage <string>] [-footerText <string>] [-loginSysMessage <string>] [-preLoginButton <string>] [-preLoginMessage <string>] [-preLoginTitle <string>] [-domainSelection <string>] [-siteType ( XenAppWeb | XenAppServices ) [-ShowSearch ( ON | OFF )] [-ShowRefresh ( ON | OFF )] [-wiUserInterfaceModes ( SIMPLE | ADVANCED )] [-UserInterfaceLayouts <UserInterfaceLayouts>]] [-userInterfaceBranding ( Desktops | Applications )] [-publishedResourceType <publishedResourceType>] [-kioskMode ( ON | OFF )] [-restrictDomains ( ON | OFF )] [-loginDomains <string>] [-hideDomainField ( ON | OFF )] [-agCallbackURL <string>]
```

Arguments

sitePath

Path to the Web Interface site being created on the NetScaler appliance.

agURL

Call back URL of the Gateway.

staURL

URL of the Secure Ticket Authority (STA) server.

secondSTAURL

URL of the second Secure Ticket Authority (STA) server.

sessionReliability

Enable session reliability through Access Gateway.

Possible values: ON, OFF

Default value: OFF

useTwoTickets

Request tickets issued by two separate Secure Ticket Authorities (STA) when a resource is accessed.

Possible values: ON, OFF

Default value: OFF

authenticationPoint

Authentication point for the Web Interface site.

Possible values: WebInterface, AccessGateway

agAuthenticationMethod

Method for authenticating a Web Interface site if you have specified Web Interface as the authentication point.

Available settings function as follows:

- * Explicit - Users must provide a user name and password to log on to the Web Interface.

- * Anonymous - Users can log on to the Web Interface without providing a user name and password. They have access to resources published for anonymous users.

Possible values: Explicit, SmartCard

wiAuthenticationMethods

The method of authentication to be used at Web Interface

Default value: Explicit

defaultCustomTextLocale

Default language for the Web Interface site.

Possible values: German, English, Spanish, French, Japanese, Korean, Russian, Chinese_simplified, Chinese_traditional

Default value: English

webSessionTimeout

Time-out, in minutes, for idle Web Interface browser sessions. If a client's session is idle for a time that exceeds the time-out value, the NetScaler appliance terminates the connection.

Default value: 20

Minimum value: 1

Maximum value: 1440

defaultAccessMethod

Default access method for clients accessing the Web Interface site.

Note: Before you configure an access method based on the client IP address, you must enable USIP mode on the Web Interface service to make the client's IP address available with the Web Interface.

Depending on whether the Web Interface site is configured to use an HTTP or HTTPS virtual server or to use access gateway, you can send clients or access gateway the IP address, or the alternate address, of a XenApp or XenDesktop server. Or, you can send the IP address translated from a mapping entry, which defines mapping of an internal address and port to an external address and port.

Note: In the NetScaler command line, mapping entries can be created by using the bind wi site command.

Possible values: Direct, Alternate, Translated, GatewayDirect, GatewayAlternate, GatewayTranslated

loginTitle

A custom login page title for the Web Interface site.

Default value: "Welcome to Web Interface on NetScaler"

appWelcomeMessage

Specifies localized text to appear at the top of the main content area of the Applications screen. LanguageCode is en, de, es, fr, ja, or any other supported language identifier.

welcomeMessage

Localized welcome message that appears on the welcome area of the login screen.

footerText

Localized text that appears in the footer area of all pages.

loginSysMessage

Localized text that appears at the bottom of the main content area of the login screen.

preLoginButton

Localized text that appears as the name of the pre-login message confirmation button.

preLoginMessage

Localized text that appears on the pre-login message page.

preLoginTitle

Localized text that appears as the title of the pre-login message page.

domainSelection

Domain names listed on the login screen for explicit authentication.

siteType

Type of access to the Web Interface site. Available settings function as follows:

- * XenApp/XenDesktop web site - Configures the Web Interface site for access by a web browser.
- * XenApp/XenDesktop services site - Configures the Web Interface site for access by the XenApp plug-in.

Possible values: XenAppWeb, XenAppServices

Default value: XenAppWeb

userInterfaceBranding

Specifies whether the site is focused towards users accessing applications or desktops. Setting the parameter to Desktops changes the functionality of the site to improve the experience for XenDesktop users. Citrix recommends using this setting for any deployment that includes XenDesktop.

Possible values: Desktops, Applications

Default value: Applications

publishedResourceType

Method for accessing the published XenApp and XenDesktop resources.

Available settings function as follows:

- * Online - Allows applications to be launched on the XenApp and XenDesktop servers.
- * Offline - Allows streaming of applications to the client.
- * DualMode - Allows both online and offline modes.

Possible values: Online, Offline, DualMode

Default value: Online

kioskMode

User settings do not persist from one session to another.

Possible values: ON, OFF

Default value: OFF

ShowSearch

Enables search option on XenApp websites

Possible values: ON, OFF

Default value: OFF

ShowRefresh

Provides the Refresh button on the applications screen.

Possible values: ON, OFF

Default value: OFF

wiUserInterfaceModes

Appearance of the login screen.

* Simple - Only the login fields for the selected authentication method are displayed.

* Advanced - Displays the navigation bar, which provides access to the pre-login messages and preferences screens.

Possible values: SIMPLE, ADVANCED

Default value: SIMPLE

UserInterfaceLayouts

Specifies whether or not to use the compact user interface.

Possible values: AUTO, NORMAL, COMPACT

Default value: AUTO

restrictDomains

The RestrictDomains setting is used to enable/disable domain restrictions. If domain restriction is enabled, the LoginDomains list is used for validating the login domain. It is applied to all the authentication methods except Anonymous for XenApp Web and XenApp Services sites

Possible values: ON, OFF

Default value: OFF

loginDomains

[List of NetBIOS domain names], Domain names to use for access restriction.

Only takes effect when used in conjunction with the RestrictDomains setting.

hideDomainField

The HideDomainField setting is used to control whether the domain field is displayed on the logon screen.

Possible values: ON, OFF

Default value: OFF

agCallbackURL

Callback AGURL to which Web Interface contacts.

Example

```
add wi site /Citrix/PNAgent -siteType XenAppServices
```

rm wi site

Removes a Web Interface site from the NetScaler appliance.

Synopsys

```
rm wi site <sitePath>
```

Arguments

sitePath

Path to the Web Interface site being created on the NetScaler appliance.

Example

```
rm wi site /Citrix/PNAgent
```

set wi site

Modifies the parameters of a Web Interface site configured on the NetScaler appliance.

Synopsys

```
set wi site <sitePath> [-agURL <string>] [-staURL <string>] [-sessionReliability ( ON | OFF )] [-useTwoTickets ( ON | OFF )] [-secondSTAURL <string>] [-wiAuthenticationMethods ( Explicit | Anonymous ) ...] [-defaultAccessMethod <defaultAccessMethod>] [-defaultCustomTextLocale <defaultCustomTextLocale>] [-webSessionTimeout <positive_integer>] [-loginTitle <string>] [-appWelcomeMessage <string>] [-welcomeMessage <string>] [-footerText <string>] [-loginSysMessage <string>] [-preLoginButton <string>] [-preLoginMessage <string>] [-preLoginTitle <string>] [-domainSelection <string>] [-userInterfaceBranding ( Desktops | Applications )] [-authenticationPoint ( WebInterface | AccessGateway )] [-agAuthenticationMethod ( Explicit | SmartCard )] [-publishedResourceType <publishedResourceType>] [-kioskMode ( ON | OFF )] [-ShowSearch ( ON | OFF )] [-ShowRefresh ( ON | OFF )] [-wiUserInterfaceModes ( SIMPLE | ADVANCED )] [-UserInterfaceLayouts <UserInterfaceLayouts>] [-restrictDomains ( ON | OFF )] [-loginDomains <string>] [-hideDomainField ( ON | OFF )] [-agCallbackURL <string>]
```

Arguments

sitePath

Path to the Web Interface site being created on the NetScaler appliance.

agURL

Call back URL of the Gateway.

staURL

URL of the Secure Ticket Authority (STA) server.

sessionReliability

Enable session reliability through Access Gateway.

Possible values: ON, OFF

Default value: OFF

useTwoTickets

Request tickets issued by two separate Secure Ticket Authorities (STA) when a resource is accessed.

Possible values: ON, OFF

Default value: OFF

secondSTAURL

URL of the second Secure Ticket Authority (STA) server.

wiAuthenticationMethods

The method of authentication to be used at Web Interface

Default value: Explicit

defaultAccessMethod

Default access method for clients accessing the Web Interface site.

Note: Before you configure an access method based on the client IP address, you must enable USIP mode on the Web Interface service to make the client's IP address available with the Web Interface.

Depending on whether the Web Interface site is configured to use an HTTP or HTTPS virtual server or to use access gateway, you can send clients or access gateway the IP address, or the alternate address, of a XenApp or XenDesktop server. Or, you can send the IP address translated from a mapping entry, which defines mapping of an internal address and port to an external address and port.

Note: In the NetScaler command line, mapping entries can be created by using the bind wi site command.

Possible values: Direct, Alternate, Translated, GatewayDirect, GatewayAlternate, GatewayTranslated

defaultCustomTextLocale

Default language for the Web Interface site.

Possible values: German, English, Spanish, French, Japanese, Korean, Russian, Chinese_simplified, Chinese_traditional

Default value: English

webSessionTimeout

Time-out, in minutes, for idle Web Interface browser sessions. If a client's session is idle for a time that exceeds the time-out value, the NetScaler appliance terminates the connection.

Default value: 20

Minimum value: 1

Maximum value: 1440

loginTitle

A custom login page title for the Web Interface site.

Default value: "Welcome to Web Interface on NetScaler"

appWelcomeMessage

Specifies localized text to appear at the top of the main content area of the Applications screen. LanguageCode is en, de, es, fr, ja, or any other supported language identifier.

welcomeMessage

Localized welcome message that appears on the welcome area of the login screen.

footerText

Localized text that appears in the footer area of all pages.

loginSysMessage

Localized text that appears at the bottom of the main content area of the login screen.

preLoginButton

Localized text that appears as the name of the pre-login message confirmation button.

preLoginMessage

Localized text that appears on the pre-login message page.

preLoginTitle

Localized text that appears as the title of the pre-login message page.

domainSelection

Domain names listed on the login screen for explicit authentication.

userInterfaceBranding

Specifies whether the site is focused towards users accessing applications or desktops. Setting the parameter to Desktops changes the functionality of the site to improve the experience for XenDesktop users. Citrix recommends using this setting for any deployment that includes XenDesktop.

Possible values: Desktops, Applications

Default value: Applications

authenticationPoint

Authentication point for the Web Interface site.

Possible values: WebInterface, AccessGateway

agAuthenticationMethod

Method for authenticating a Web Interface site if you have specified Web Interface as the authentication point.

Available settings function as follows:

- * Explicit - Users must provide a user name and password to log on to the Web Interface.

- * Anonymous - Users can log on to the Web Interface without providing a user name and password. They have access to resources published for anonymous users.

Possible values: Explicit, SmartCard

publishedResourceType

Method for accessing the published XenApp and XenDesktop resources.

Available settings function as follows:

- * Online - Allows applications to be launched on the XenApp and XenDesktop servers.

- * Offline - Allows streaming of applications to the client.

- * DualMode - Allows both online and offline modes.

Possible values: Online, Offline, DualMode

Default value: Online

kioskMode

User settings do not persist from one session to another.

Possible values: ON, OFF

Default value: OFF

ShowSearch

Enables search option on XenApp websites

Possible values: ON, OFF

Default value: OFF

ShowRefresh

Provides the Refresh button on the applications screen.

Possible values: ON, OFF

Default value: OFF

wiUserInterfaceModes

Appearance of the login screen.

* Simple - Only the login fields for the selected authentication method are displayed.

* Advanced - Displays the navigation bar, which provides access to the pre-login messages and preferences screens.

Possible values: SIMPLE, ADVANCED

Default value: SIMPLE

UserInterfaceLayouts

Specifies whether or not to use the compact user interface.

Possible values: AUTO, NORMAL, COMPACT

Default value: AUTO

restrictDomains

The RestrictDomains setting is used to enable/disable domain restrictions. If domain restriction is enabled, the LoginDomains list is used for validating the login domain. It is applied to all the authentication methods except Anonymous for XenApp Web and XenApp Services sites

Possible values: ON, OFF

Default value: OFF

loginDomains

[List of NetBIOS domain names], Domain names to use for access restriction.

Only takes effect when used in conjunction with the RestrictDomains setting.

hideDomainField

The HideDomainField setting is used to control whether the domain field is displayed on the logon screen.

Possible values: ON, OFF

Default value: OFF

agCallbackURL

Callback AGURL to which Web Interface contacts.

Example

```
set wi site /Citrix/PNAgent -staURL http://myStaServer
```

unset wi site

Use this command to remove wi site settings. Refer to the set wi site command for meanings of the arguments.

Synopsys

```
unset wi site <sitePath> [-appWelcomeMessage] [-welcomeMessage] [-footerText] [-loginSysMessage] [-preLoginButton] [-preLoginMessage] [-preLoginTitle] [-userInterfaceBranding] [-loginDomains]
```

bind wi site

Binds XenApp or XenDesktop farms to a Web Interface site and optionally, defines access methods for different client IP addresses or networks.

Synopsys

```
bind wi site <sitePath> ((<farmName> <xmlServerAddresses> [-groups <string>] [-recoveryFarm ( ON | OFF )] [-xmlPort <positive_integer>] [-transport <transport> [-sslRelayPort <positive_integer>]] [-loadBalance ( ON | OFF )]) | ((-accessMethod <accessMethod> (-clientIpAddress <ip_addr> -clientNetMask <netmask>)) | (-translationInternalIp <ip_addr> -translationInternalPort <port|*> -translationExternalIp <ip_addr> -translationExternalPort <port|*> [-accessType <accessType>])))
```

Arguments

sitePath

Path to the Web Interface site.

farmName

Name for the logical representation of a XenApp or XenDesktop farm to be bound to the Web Interface site. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

xmlServerAddresses

Comma-separated IP addresses or host names of XenApp or XenDesktop servers providing XML services.

groups

Active Directory groups that are permitted to enumerate resources from server farms. Including a setting for this parameter activates the user roaming feature. A maximum of 512 user groups can be specified for each farm defined with the Farm<n> parameter. The groups must be comma separated.

recoveryFarm

Binded farm is set as a recovery farm.

Possible values: ON, OFF

Default value: OFF

xmlPort

Port number at which to contact the XML service.

Default value: 80

Minimum value: 1

Maximum value: 65535

transport

Transport protocol to use for transferring data, related to the Web Interface site, between the NetScaler appliance and the XML service.

Possible values: HTTP, HTTPS, SSLRELAY

Default value: HTTP

sslRelayPort

TCP port at which the XenApp or XenDesktop servers listen for SSL Relay traffic from the NetScaler appliance. This parameter is required if you have set SSL Relay as the transport protocol.

Web Interface uses root certificates when authenticating a server running SSL Relay. Make sure that all the servers running SSL Relay are configured to listen on the same port.

Default value: 443

Minimum value: 1

Maximum value: 65535

loadBalance

Use all the XML servers (load balancing mode) or only one server (failover mode).

Possible values: ON, OFF

Default value: ON

accessMethod

Secure access method to be applied to the IPv4 or network address of the client specified by the Client IP Address parameter.

Depending on whether the Web Interface site is configured to use an HTTP or HTTPS virtual server or to use access gateway, you can send clients or access gateway the IP address, or the alternate address, of a XenApp or XenDesktop server. Or, you can send the IP address translated from a mapping entry, which defines mapping of an internal address and port to an external address and port.

Possible values: Direct, Alternate, Translated, GatewayDirect, GatewayAlternate, GatewayTranslated

clientIpAddress

IPv4 or network address of the client for which you want to associate an access method.

Default value: 0

clientNetMask

Subnet mask associated with the IPv4 or network address specified by the Client IP Address parameter.

Default value: 0

translationInternalIp

IP address of the server for which you want to associate an external IP address. (Clients access the server through the associated external address and port.)

Default value: 0

translationInternalPort

Port number of the server for which you want to associate an external port. (Clients access the server through the associated external address and port.)

translationExternalIp

External IP address associated with server's IP address.

translationExternalPort

External port number associated with the server's port number.

accessType

Type of access to the XenApp or XenDesktop server.

Available settings function as follows:

* User Device - Clients can use the translated address of the mapping entry to connect to the XenApp or XenDesktop server.

* Gateway - Access Gateway can use the translated address of the mapping entry to connect to the XenApp or XenDesktop server.

* User Device and Gateway - Both clients and Access Gateway can use the translated address of the mapping entry to connect to the XenApp or XenDesktop server.

Possible values: UserDevice, Gateway, UserDeviceAndGateway

Default value: UserDevice

Example

```
bind wi site /Citrix/XenApp Farm2 10.10.10.11
```

unbind wi site

Unbinds XenApp or XenDesktop farms from the Web Interface site and removes the existing access method definition for a client IP address or network.

Synopsys

```
unbind wi site <sitePath> (<farmName> | ((-clientIpAddress <ip_addr> -clientNetMask <netmask>) | (-translationInternalIp <ip_addr> -translationInternalPort <port|*> -translationExternalIp <ip_addr> -translationExternalPort <port|*>)))
```

Arguments

sitePath

Path to the Web Interface site.

farmName

Name of the XenApp farm to be unbound from the Web Interface site.

clientIpAddress

IPv4 address or network address of the client for which you want to remove the defined access method.

Default value: 0

clientNetMask

Subnet mask associated with the IPv4 or network address.

Default value: 0

translationInternalIp

Internal IP address of a mapping entry to be removed.

Default value: 0

translationInternalPort

Internal port of a mapping entry to be removed.

translationExternalIp

External IP address of a mapping entry to be removed. (The external IP address is mapped to an internal address.)

translationExternalPort

External port of a mapping entry to be removed. (The external port is mapped to an internal port.)

Example

```
unbind wi site /Citrix/XenApp Farm2
```

show wi site

Displays settings of all the Web Interface sites, or of a specified site. To display settings of all the Web Interface sites, run the command without any parameters.

Synopsys

show wi site [<sitePath>]

Arguments

sitePath

Path of a Web Interface site whose details you want the NetScaler appliance to display.

Outputs

stateflag

agURL

Call back URL of the Gateway.

staURL

The URL of Secure Ticketing Authority server

wiAuthenticationMethods

The method of authentication to be used at Web Interface

loginTitle

A custom login page title for the Web Interface site.

appWelcomeMessage

Specifies localized text to appear at the top of the main content area of the Applications screen. LanguageCode is en, de, es, fr, ja, or any other supported language identifier.

welcomeMessage

Localized welcome message that appears on the welcome area of the login screen.

footerText

Localized text that appears in the footer area of all pages.

loginSysMessage

Localized text that appears at the bottom of the main content area of the login screen.

preLoginButton

Localized text that appears as the name of the pre-login message confirmation button.

preLoginMessage

Localized text that appears on the pre-login message page.

preLoginTitle

Localized text that appears as the title of the pre-login message page.

domainSelection

Domain names listed on the login screen for explicit authentication.

defaultCustomTextLocale

Default language for the Web Interface site.

webSessionTimeout

Time-out, in minutes, for idle Web Interface browser sessions. If a client's session is idle for a time that exceeds the time-out value, the NetScaler appliance terminates the connection.

siteType

Type of access to the Web Interface site. Available settings function as follows:

- * XenApp/XenDesktop web site - Configures the Web Interface site for access by a web browser.
- * XenApp/XenDesktop services site - Configures the Web Interface site for access by the XenApp plug-in.

userInterfaceBranding

Specifies whether the site is focused towards users accessing applications or desktops. Setting the parameter to Desktops changes the functionality of the site to improve the experience for XenDesktop users. Citrix recommends using this setting for any deployment that includes XenDesktop.

ShowSearch

Enables search option on XenApp websites

ShowRefresh

Provides the Refresh button on the applications screen.

wiUserInterfaceModes

Appearance of the login screen.

- * Simple - Only the login fields for the selected authentication method are displayed.
- * Advanced - Displays the navigation bar, which provides access to the pre-login messages and preferences screens.

UserInterfaceLayouts

Specifies whether or not to use the compact user interface.

publishedResourceType

Method for accessing the published XenApp and XenDesktop resources.

Available settings function as follows:

- * Online - Allows applications to be launched on the XenApp and XenDesktop servers.
- * Offline - Allows streaming of applications to the client.
- * DualMode - Allows both online and offline modes.

defaultAccessMethod

Default access method for clients accessing the Web Interface site.

Note: Before you configure an access method based on the client IP address, you must enable USIP mode on the Web Interface service to make the client's IP address available with the Web Interface.

Depending on whether the Web Interface site is configured to use an HTTP or HTTPS virtual server or to use access gateway, you can send clients or access gateway the IP address, or the alternate address, of a XenApp or XenDesktop server. Or, you can send the IP address translated from a mapping entry, which defines mapping of an internal address and port to an external address and port.

Note: In the NetScaler command line, mapping entries can be created by using the bind wi site command.

farmName

Name for the logical representation of a XenApp or XenDesktop farm to be bound to the Web Interface site. Must begin with an ASCII alphabetic or underscore (_) character, and must contain only ASCII alphanumeric, underscore, hash (#), period (.), space, colon (:), at (@), equals (=), and hyphen (-) characters.

accessMethod

Secure access method to be applied to the IPv4 or network address of the client specified by the Client IP Address parameter.

Depending on whether the Web Interface site is configured to use an HTTP or HTTPS virtual server or to use access gateway, you can send clients or access gateway the IP address, or the alternate address, of a XenApp or XenDesktop server. Or, you can send the IP address translated from a mapping entry, which defines mapping of an internal address and port to an external address and port.

clientIpAddress

IPv4 or network address of the client for which you want to associate an access method.

clientNetMask

Subnet mask associated with the IPv4 or network address specified by the Client IP Address parameter.

translationInternalIp

IP address of the server for which you want to associate an external IP address. (Clients access the server through the associated external address and port.)

translationInternalPort

Port number of the server for which you want to associate an external port. (Clients access the server through the associated external address and port.)

translationExternalIp

External IP address associated with server's IP address.

translationExternalPort

External port number associated with the server's port number.

accessType

Type of access to the XenApp or XenDesktop server.

Available settings function as follows:

- * User Device - Clients can use the translated address of the mapping entry to connect to the XenApp or XenDesktop server.
- * Gateway - Access Gateway can use the translated address of the mapping entry to connect to the XenApp or XenDesktop server.
- * User Device and Gateway - Both clients and Access Gateway can use the translated address of the mapping entry to connect to the XenApp or XenDesktop server.

xmlServerAddresses

Comma-separated IP addresses or host names of XenApp or XenDesktop servers providing XML services.

xmlPort

Port number at which to contact the XML service.

transport

Transport protocol to use for transferring data, related to the Web Interface site, between the NetScaler appliance and the XML service.

sslRelayPort

TCP port at which the XenApp or XenDesktop servers listen for SSL Relay traffic from the NetScaler appliance. This parameter is required if you have set SSL Relay as the transport protocol.

Web Interface uses root certificates when authenticating a server running SSL Relay. Make sure that all the servers running SSL Relay are configured to listen on the same port.

agAuthenticationMethod

Method for authenticating a Web Interface site if you have specified Web Interface as the authentication point.

Available settings function as follows:

- * Explicit - Users must provide a user name and password to log on to the Web Interface.

- * Anonymous - Users can log on to the Web Interface without providing a user name and password. They have access to resources published for anonymous users.

groups

Active Directory groups that are permitted to enumerate resources from server farms. Including a setting for this parameter activates the user roaming feature. A maximum of 512 user groups can be specified for each farm defined with the Farm<n> parameter. The groups must be comma separated.

recoveryFarm

Binded farm is set as a recovery farm.

sessionReliability

Enable session reliability through Access Gateway.

useTwoTickets

Request tickets issued by two separate Secure Ticket Authorities (STA) when a resource is accessed.

secondSTAURL

URL of the second Secure Ticket Authority (STA) server.

loadBalance

Use all the XML servers (load balancing mode) or only one server (failover mode).

authenticationPoint

Authentication point for the Web Interface site.

kioskMode

User settings do not persist from one session to another.

restrictDomains

The RestrictDomains setting is used to enable/disable domain restrictions. If domain restriction is enabled, the LoginDomains list is used for validating the login domain. It is applied to all the authentication methods except Anonymous for XenApp Web and XenApp Services sites

loginDomains

[List of NetBIOS domain names], Domain names to use for access restriction.

Only takes effect when used in conjunction with the RestrictDomains setting.

hideDomainField

The HideDomainField setting is used to control whether the domain field is displayed on the logon screen.

agCallbackURL

Callback AGURL to which Web Interface contacts.

devno

count

Example

```
show wi site
```


Quick Start Guides

A reference to quick installation and configuration of your hardware appliance.

Title
Citrix NetScaler Quick Start Guide for MPX 24100, 24150
Citrix NetScaler Quick Start Guide for MPX 25100T, 25160T

Glossary

[A](#) | [B](#) | [C](#) | [D](#) | [E](#) | [F](#) | [G](#) | [H](#) | [I](#) | [L](#) | [M](#) | [N](#) | [O](#) | [P](#) | [R](#) | [S](#) | [T](#) | [U](#) | [V](#) | [W](#) | [X](#)

A

AAA

A NetScaler feature providing authentication, authorization, and auditing for all application traffic.

AAA-TM

See AAA. The configuration utility displays AAA as AAA-TM, meaning AAA traffic management.

access control

A general term denoting something that controls access to a resource. A more specific term is usually preferable.

Access Gateway

Former name of NetScaler Gateway.

action

A policy element that specifies what to do with a request or response that matches the expression in the policy. For example, if an expression in a policy matches a particular source IP address in a request, the action associated with the policy determines whether the connection is permitted.

action analytics

A NetScaler data-collection feature that can automatically optimize traffic in real time.

active-active mode

A deployment mode that, in addition to preventing downtime, makes efficient use of all the NetScaler ADCs in the deployment. In active-active deployment mode, the same virtual IP (VIP) addresses are assigned to all NetScaler ADCs in the configuration, but with different priorities, so that a given VIP can be active on only one ADC at a time. The ADCs can be configured so that no NetScaler ADC is idle.

ADC

See application delivery controller.

Amazon Elastic Block Store (EBS)

AWS feature that provides storage volumes that can be attached to EC2 instances.

Amazon Machine Image (AMI)

A special type of virtual appliance used to instantiate (create) a virtual machine within the Amazon Elastic Compute Cloud (EC2). It serves as the basic unit of deployment for services delivered through EC2.

AMI

See Amazon Machine Image.

AppFlow

A NetScaler feature that provides transaction-level visibility into HTTP, SSL, TCP, and SSL_TCP traffic flows.

application delivery controller (ADC)

A product, such as Citrix NetScaler, that optimizes delivery of applications. An ADC provides advanced features in addition to basic load balancing.

application visualizer

A graphical representation of an AppExpert application. Displays the public endpoints, application units, backend services, and policies that are configured for the application. You can use the Visualizer to obtain a visual overview of an AppExpert application's configuration and configure some of the displayed entities. By default, the Visualizer displays application units, services, and monitors for the selected application.

AVP

See Attribute-Value Pair.

Attribute-Value Pair (AVP)

AVPs are the basic units inside a Diameter and Radius message that carry authentication, security, and any other data pertaining to the application. There must be at least one AVP inside a Diameter or RADIUS message.

auditing

Feature that keeps a record of each user's activity on a protected server.

authentication

Feature for verifying client credentials, either locally or with a third-party authentication server, and allowing only approved users to access protected servers.

authorization

Feature for verifying which content on a protected server each user is allowed to access.

AWS region

Amazon EC2 is hosted in multiple locations world-wide. These locations are composed of regions and Availability Zones. Each region is a separate geographic area. Each region has multiple, isolated locations known as Availability Zones. Amazon EC2 gives you the ability to place resources, such as instances, and data in multiple locations. Resources are not replicated across regions unless you do so specifically.

B

back end

The server-facing side of a network.

bind point

An entity, or a stage of traffic processing, at which traffic is examined to see if it matches a policy. For example, a bind point can be a load balancing virtual server, or it can apply to all traffic at given stage of processing, such as when a request is received or when a response is sent.

bridge group

A NetScaler feature for merging multiple VLANs into a single broadcast domain.

C

cache redirection

A policy and virtual-server based NetScaler feature that evaluates the type of content requested and directs requests to a cache instead of a server.

call home

A NetScaler feature that monitors the appliance and automatically uploads data to the Support server if an error condition is detected.

CEA

Capabilities Exchange Answer. A message used by the Diameter protocol to establish a connection.

CER

Capabilities Exchange Request. A message used by the Diameter protocol to establish a connection.

certificate-key pair

An SSL certificate and its corresponding private key. Stored on a NetScaler ADC that offloads SSL processing from servers.

classic policy

The older, less robust type of NetScaler policy.

CLI

Command-line interface.

client

A computer that receives data from a server. Can also refer to a person.

client data plane

The logical grouping of the physical connections between the cluster nodes and the client-side connecting device.

client drive mapping

A method that enables users to access some or all of their clients'™ drives from an application running on an application server.

client keep-alive

A setting that enables receiving multiple client requests on a single client connection. Applies only to HTTP and HTTPS services.

CloudBridge Connector

A NetScaler feature for connecting a datacenter to a cloud or another datacenter. The feature establishes a "CloudBridge Connector tunnel" between the connected entities.

CloudFormation

An Amazon Web Services (AWS) feature for managing cloud resources. Uses templates to manage groups of resources, which are called "stacks."

CloudPlatform

A Citrix software platform (powered by Apache CloudStack) that pools computing resources to build public, private, and hybrid Infrastructure as a Service (IaaS) clouds.

cluster

A group of nCore appliances working together as a single system image. The cluster can include as few as 2 or as many as 32 NetScaler nCore hardware or virtual appliances as nodes. The client traffic is distributed between the nodes to provide high availability, high throughput, and scalability.

cluster backplane

The logical grouping of the physical connections between the cluster nodes and the cluster backplane switch. The nodes of a cluster communicate with each other over the cluster backplane.

cluster backplane switch

A switch through which cluster nodes communicate with each other.

cluster instance

A logical entity created on the first node added to a NetScaler cluster. The cluster instance is assigned a cluster ID, which uniquely identifies the cluster.

cluster IP address

The management IP address of a cluster, through which configuration tasks must be performed. This IP address is owned by the cluster's configuration coordinator.

cluster link aggregation

A feature combining groups of cluster interfaces into channels. Similar to NetScaler link aggregation, but without the requirement that all interfaces be on the same appliance. The interfaces can be on different nodes of the same cluster.

cluster node

A NetScaler ADC that is part of a cluster.

cluster propagation

The process through which configurations that are performed on the cluster IP address are propagated to all the nodes of the cluster.

cluster synchronization

The process through which cluster configurations are synchronized to appliances that are added as cluster nodes or to nodes that rejoin the cluster.

clustering

The process of creating a cluster of NetScaler ADCs.

collector

An AppFlow entity that receives flow records generated by a NetScaler appliance.

community string

A password for authenticating SNMP queries from SNMP managers.

configuration coordinator

The cluster node on which cluster configurations are performed and then propagated to the other cluster nodes. The configuration coordinator owns the cluster IP address.

configuration utility

The NetScaler graphical user interface (GUI).

console

The command line interface accessed through the console port of a NetScaler appliance.

content filtering

A NetScaler feature that takes a user-specified action when the something in the header of a request or response matches a policy.

content group

A group of all virtual servers and policies involved in a particular content switching configuration.

content switching

Load balancing that bases server selection on the type of content requested.

critical interface

A NetScaler interface that, if it fails or is disabled, triggers a high-availability (HA) failover.

crossover cable

An Ethernet cable in which the sending and receiving wires are crossed.

D

dashboard

An interface element that displays performance data in graphical form.

data set

A specialized form of pattern set, consisting of an array of patterns of type number (integer), IPv4 address, or IPv6 address.

datacenter

A facility housing computer systems and associated components, such as telecommunications and storage systems. Usually includes redundant or backup power supplies, redundant data communications connections, environmental controls such as air conditioning and fire suppression, and security devices.

DataStream

NetScaler feature for load balancing database servers.

default syntax policies

The newer type of NetScaler policies, which provide more capabilities than do classic policies. Most NetScaler features are migrating from classic to default syntax policies.

Desktop Director

A Citrix product that provides a detailed and intuitive overview of XenDesktop environments.

Diameter

An Authentication, Authorization, and Accounting (AAA) protocol derived from RADIUS.

direct server return (DSR)

A NetScaler load balancing mode in which servers send responses directly to the clients, instead of through the NetScaler ADC.

Disconnect Peer Acknowledgment (DPA)

A response acknowledging a DPR.

Disconnect Peer Request (DPR)

A Diameter request sent to a peer to initiate session termination.

down state flush

A NetScaler feature for delayed cleanup of a virtual-server's connections. Connections remain open until the virtual server enters the DOWN state.

DPA

See Disconnect Peer Acknowledgment.

DPR

See Disconnect Peer Request.

DSR

See direct server return.

E

EBS

See Amazon Elastic Block Store.

EC2

See Elastic Cloud Compute.

effective state

Cumulative state of the primary and backup virtual servers. If any of virtual servers in the chain is UP, the effective state is UP. For a GSLB service, the effective state reflects the effective state of the corresponding load balancing virtual server. (A load balancing *service* has no effective state.)

Elastic Cloud Compute (EC2)

Amazon Web Services (AWS) feature with which users create virtual computers (instances).

Elastic Network Interface (ENI)

An Amazon Web Services (AWS) virtual network interface that you can attach to an instance in a virtual private cloud (VPC).

ENI

See Elastic Network Interface.

ETag

An identifier assigned by a web server to a specific resource at a URL. Useful for cache validation and for preventing one simultaneous update of a resource from overwriting another.

expression

A logic statement, such as a Perl Compatible Regular Expression (PCRE), specifying the characteristics of requests or responses that match the policy of which the expression is a part. Also called a rule.

F

failover interface set

A pair of interfaces, with each interface on a different appliance. If one appliance fails, process is transferred to the other appliance without triggering a failover event.

Federal Information Processing Standards (FIPS)

Standards developed by the National Institute for Standards and Technology (NIST) to ensure compliance with federal security and data-privacy requirements.

field replaceable unit (FRU)

A NetScaler component that can be replaced by the user.

FIPS

See Federal Information Processing Standards.

FRU

See field replaceable unit.

flow processor

The cluster node that is selected as the node to process the traffic. The flow processor receives the traffic from the flow receiver through the cluster backplane.

flow receiver

The cluster node that receives traffic from the external network. The flow receiver node steers or forwards the traffic to the flow processor through the cluster backplane.

front end

The client-facing side of a network.

G

global server load balancing (GSLB)

A NetScaler feature that performs load balancing across data centers in a WAN.

GSLB

See global server load balancing.

GUI

See configuration utility.

H

HDX Insight

NetScaler Insight Center component that monitors ICA traffic.

high availability (HA)

A deployment mode in which one appliance (the primary) is backed up by another appliance (the secondary). If the primary appliance fails, a failover event transfers control to the secondary appliance.

HA

See high availability.

I

IAM

See Identity and Access Management.

INC

See Independent Network Configuration (INC) mode.

Identity and Access Management (IAM)

An Amazon Web Services (AWS) feature with which you can control access to AWS services and resources for your users. Using IAM, you can create and manage AWS users and groups and use permissions to allow or deny their access to AWS resources.

ICA

See Independent Computing Architecture (ICA)

Independent Computing Architecture (ICA)

Citrix proprietary protocol for XenApp and XenDesktop traffic.

Independent Network Configuration (INC) mode

A type of High availability deployment in which the two HA nodes reside in different networks. The following independent network entities and configurations are neither propagated nor synced to the other node: MIPs, SNIPs, VLANs, routes (except LLB routes), route monitors, RNAT rules (except any RNAT rule with a VIP as the NAT IP), and dynamic routing configurations.

information element

A description of an attribute that can appear in an IPFIX Record. RFC5102 defines the base set of IPFIX information elements.

inline mode

A two-arm deployment mode in which the traffic between clients and servers passes through the deployed appliance.

IP set

A set of subnet IP (SNIP) addresses and virtual IP (VIP) addresses, identified with a meaningful name indicating the usage of the IP addresses contained in the set.

L

link load balancing (LLB)

A NetScaler feature that balances outbound traffic across multiple Internet connections provided by different service providers.

Linkset

An entity specifying interfaces through which a node can connect to the external switch through the cluster backplane. Linksets must be used for traffic distribution in an asymmetric (some nodes not connected to the external switch) cluster topology. Linksets can be used exclusively or combined with ECMP or cluster link aggregation.

load balancing

A core NetScaler feature that distributes user requests for web pages and other protected applications across multiple servers that all host (or mirror) the same content.

load balancing virtual server

The IP address, port, and protocol combination to which a client sends connection requests for a particular load-balanced website or application. If the application is accessible from the Internet, the virtual server IP (VIP) address is a public IP address. If the application is accessible only from the local area network (LAN) or wide area network (WAN), the VIP is usually a private (ICANN non-routable) IP address.

M

management service

The graphical user interface (GUI) of a NetScaler SDX appliance. Also used on some CloudBridge appliances.

mapped IP (MIP)

Mapped IP address. A NetScaler IP address used for server-side connections. Citrix recommends using a subnet IP (SNIP) address instead.

MEP

Metric Exchange Protocol, a Citrix protocol used for GSLB.

MIP

See mapped IP.

MIR

See multiple IP response.

monitor

A NetScaler entity that periodically probes each service to which you assign it. If a service does not respond within a specified interval, and the specified number of probes fail, the service is marked DOWN. In that case, the monitor continues to send probes, and can change the status to UP.

multiple IP response (MIR)

A GSLB option for the NetScaler ADC to send multiple IP addresses in response to a DNS request.

multitenant

A configuration in which multiple clients are served from the same platform.

N

named expression

An expression that has been assigned a name, which is used instead of the expression itself in a policy.

nCore

The multiple-core, 64-bit version of the NetScaler operating system.

negative caching

The caching of negative responses from servers in a domain, to speed up responses to queries.

net profile

An information set that contains NetScaler owned IP addresses (IP set) or an IP address. During communication with physical servers or peers, the NetScaler ADC uses the addresses specified in the profile as source IP addresses.

netbridge

A logical container that holds or represents a CloudBridge Connector tunnel configuration on a NetScaler appliance. A GRE tunnel entity is associated with the netbridge. A particular CloudBridge Connector tunnel configuration on a NetScaler appliance is identified by the name of the netbridge entity.

netmask

A network mask. Also called a subnet mask.

NetScaler Gateway

A NetScaler feature (also available as a standalone appliance) that provides secure access to a LAN or WAN from any location on the Internet.

NetScaler Insight Center

A Citrix virtual appliance that can monitor NetScaler appliances.

NetScaler IP (NSIP)

NetScaler IP address. The IP address for management and general system access to the NetScaler appliance.

NetScaler owned IP address

An IP address that exists only on a NetScaler ADC. NetScaler owned IP addresses can be of the following types: NetScaler IP address (NSIP), Virtual IP addresses (VIPs), Subnet IP addresses (SNIPs), and global server load balancing site IP addresses (GSLBIPs). The NetScaler IP address (NSIP) uniquely identifies the NetScaler ADC on your network, and it provides access to the ADC. A Virtual IP address (VIP) is a public IP address to which a client sends requests. The NetScaler ADC terminates the client connection at the VIP and initiates a connection with a server. This new connection uses a subnet IP address (SNIP) as the source IP address for packets forwarded to the server. If you have multiple data centers that are geographically distributed, each data center can be identified by a unique global server load balancing site IP address (GSLBIP).

NetScaler software

The NetScaler operating system.

NetScaler VPX

A NetScaler virtual machine image that can be installed on a virtualization platform.

NetScaler Web Logging (NSWL) client

Software installed on the client system to collect logs of HTTP and HTTPS requests.

network visualizer

A Network feature that displays the network configuration of a NetScaler ADC, including the network configuration of the nodes in a high availability (HA) deployment. You can also modify the configuration of VLANs, interfaces, channels, and bridge groups, and perform HA configuration tasks.

Next Secure (NSEC)

A DNS record showing that no records exist between two points.

NITRO

The NetScaler API suite.

node group

A group of cluster nodes that have a specific set of cluster configurations. Node groups are used to define partially striped configurations.

node instance

A logical entity on a cluster node. The node instance is assigned a node ID, which uniquely identifies the node.

non-INC mode

A high availability-deployment mode in which both HA nodes are in the same network.

NSIP

See NetScaler IP.

NSVLAN

The virtual LAN (VLAN) to which the subnet that includes the NetScaler management IP (NSIP) address is bound. This subnet is available only on interfaces that are associated with NSVLAN.

O

one-arm mode

Configuration in which only one NetScaler interface is connected to an Ethernet segment.

P

partially striped configuration

A configuration available on a subset of cluster nodes. The subset is defined by the node group.

pattern set

An array of indexed patterns used for string matching during default syntax policy evaluation. Example of a pattern set: image types {svg, bmp, png, gif, tiff, jpg}. Also called a patset.

policy

An entity that identifies requests or responses on which to perform specified actions. A policy is essentially of the form if <expression>, do <action>

policy binding

The act of binding a policy to a bind point, which determines the instant at which the policy is invoked. A policy can be bound to a virtual server or globally to the NetScaler appliance.

pre-shared key

A text string manually configured on CloudBridge peers. The strings are matched against each other for authentication before security associations are established.

profile

A collection of settings that enable a feature to perform a complex function. For example, in the application firewall, a profile for XML data can perform multiple screening operations, such as examining the data for illegal XML syntax or evidence of SQL injection.

R

rate limit identifier

A named entity that specifies numeric rate-limiting thresholds, such as the maximum number of requests or connections (of a particular type) that are permitted in a specified period called a *time slice*.

rate limiting

A NetScaler feature with which you can configure the appliance to monitor the rate of traffic associated with an entity and take preventive action, in real time, when the rate reaches a specified value.

reboot

To restart an appliance.

redirection

Sending a client request to a different web page or server.

Request Switching

Citrix technology that enables an appliance to multiplex and offload TCP connections, maintain persistent connections, and manage traffic at the request (application layer) level.

responder

NetScaler feature that sends an automatic response based on who sent the request, where it is from, and other criteria with security and system-management implications.

REST interface

REpresentational State Transfer interface. A software architecture for using simple HTML calls to create or modify information on a server.

rewrite

NetScaler feature that modifies information in the headers or bodies of requests or responses.

rule

A policy element consisting of a logical expression used to evaluate requests or responses. If the evaluation returns TRUE, the action that is bound to the policy is performed.

S

SDX

An advanced NetScaler or CloudBridge platform hosting virtual machines (VMs). NetScaler SDX hosts multiple NetScaler VMs. CloudBridge SDX hosts a NetScaler VM and multiple CloudBridge VMs.

Secure Ticket Authority (STA)

The XenApp/XenDesktop entity responsible for issuing session tickets in response to connection requests for published applications on XenApp and published desktops on XenDesktop. These session tickets form the basis of authentication and authorization for access to published resources.

selectlets

A group of non-compound, default syntax expressions, each of which is called a selectlet. A traffic stream selector can contain up to five selectlets. Each selectlet is considered to be in an AND relationship with the other expressions.

selector

A filter for identifying requests, or for identifying objects in a content group.

server data plane

The logical grouping of the physical connections between cluster nodes and the server-side connecting device.

server object

A virtual entity representing a physical server. Enables naming a server, rather than identifying it by its IP address.

service

The IP address, port, and protocol combination used to route requests to a specific load-balanced application server. A service can be a logical representation of the application server itself, or of an application running on a server that hosts multiple applications. After creating a service, you bind it to a virtual server.

sessionless load balancing

Load balancing on a per packet basis, without storing session information. Reduces NetScaler resource requirements Used in DSR mode.

shell

Refers to the BSD command shell unless otherwise stated.

single sign-on (SSO)

A method of providing access to multiple password-protected information systems without requiring multiple authentications. Supported by the Citrix password manager.

single-hop mode

The mode in which NetScaler Insight Center collects data from NetScaler appliances handling connections in which users connect to XenApp and XenDesktop applications through a NetScaler Gateway appliance.

slow start

A NetScaler feature that avoids assigning all new connections to a server when it is added to the network.

spotted configuration

A configuration that is available on only a single cluster node.

SSO profile

In forms-based single signon (SSO), the profile that defines how to handle an authentication request that matches the associated policy.

start

To start an appliance (formerly "boot").

stream selector

A filter for identifying an entity for which you want to throttle access.

string map

A NetScaler entity, consisting of key-value pairs, that can be used for pattern matching in all NetScaler features that use the default policy syntax.

striped configuration

A configuration available on all nodes of a cluster.

subnet IP (SNIP)

Subnet IP address. A NetScaler-owned IP address used for server-side connections.

SureConnect

A NetScaler feature that directs requests to an alternative web page if the primary page is DOWN.

Sysid

See system ID

SYSLOG

A standard logging protocol, implemented on a SYSLOG auditing module, (which runs on the monitored appliance), and a SYSLOG server, which can run on a remote system. SYSLOG uses User Data Protocol (UDP) for data transfer

system ID (sysid)

A number, or possibly characters, identifying an appliance or virtual appliance.

T

thick provisioned format

VMDK format in which all physical disk space required for a virtual disk is allocated and zeroed out (wiped) when the disk is first created. In other words, space for a thick provisioned VMDK is reserved in advance for that VMDK only. You cannot overallocate physical disk space if you use thick provisioned VMDKs, which limits the number and size of VMDKs and can waste physical disk space, but ensures that you will have enough physical disk space in all circumstances.

thin provisioned format

VMDK format in which physical disk space needed for a virtual disk is allocated and zeroed out (wiped) only when the VMDK is written to the physical disk. In other words, space for a thin provisioned VMDK is allocated dynamically as needed; physical disk space is not allocated and reserved in advance. You can overallocate physical disk space if you use thin provisioned VMDKs, which allows you to put more and larger VMDKs on a given physical disk and avoid wasting unused space, but risks running out of physical disk space if your VMDKs are too full.

time stamp

Data indicating when an event occurred.

timeout

A setting indicating when an entity is to become unavailable (for example, how long a connection can remain idle without being closed). Also, the act of becoming unavailable after the specified period of time.

tracing

The use of trace files to debug problems in the flow of packets to the cluster nodes. The NetScaler operating system includes a utility called nstrace, which provides a dump of the packets received and sent by the appliance, and stores the packets in trace files. You use the Wireshark application to view the trace files.

traffic domains

A NetScaler feature with which you can create multiple isolated environments within a the appliance. An application belonging to a specific traffic domain communicates with entities and processes traffic within that domain. You can, for example, use the same IP address in different domains.

transparent mode

An operational mode in which an appliance between the end points of a connection does not have its own IP address. The appliance intercepts packets that one end point sends to the other.

two-arm mode

A deployment mode in which two network interfaces on the deployed appliance are connected to different Ethernet segments.

U

Use Source IP (USIP)

A NetScaler mode in which the ADC uses the client's IP address, instead of a SNIP address, in packets sent to the server.

USNIP

NetScaler mode that uses a subnet IP (SNIP) address as the source IP address of packets sent to the server, and as the address at which packets are received from the server. This mode is enabled by default.

V

view based access control model (VACM)

An SNMPv3 feature that enables you to configure access rights to a specific subtree of the MIB on the basis of various parameters, such as security level, security model, user name, and view type. You can configure agents to provide different levels of MIB access to different managers.

virtual IP (VIP)

A virtual IP (VIP) address is the IP address associated with a virtual server. It is the IP address to which clients connect for access to one of the servers represented by the virtual server. An appliance managing a wide range of traffic might have many virtual servers, each configured with its own VIP address. Some of the attributes of a VIP address are customized to meet the requirements of the virtual server.

virtual machine disk (VMDK)

File format for virtual disk drives. Originally developed by VMWare, but now an open format that is widely used in many types of clouds and virtual machines (VMs).

VMDK

See virtual machine disk

virtual server

A NetScaler entity with an IP address to which clients send requests. Distributes the requests to physical servers.

VPX

See NetScaler VPX.

W

waterfall chart

A NetScaler Insight Center chart that shows the cumulative effect of sequentially introduced positive or negative values.

Web 2.0 push

A NetScaler feature in which the NetScaler ADC functions as a proxy server to offload long-lived client TCP connections and maintain relatively fewer, reusable connections to the server.

Web Insight

A component of NetScaler Insight Center. Monitors HTTP traffic.

Web Interface

The XenApp/XenDesktop component with which users access their applications, content, and desktops through a web browser. Also supports access through the Citrix XenApp plug-in. A NetScaler ADC can manage Web Interface access to XenApp or XenDesktop server farms.

web server logging

A NetScaler feature that sends logs of HTTP and HTTPS requests to a client system for storage and retrieval.

wildcard virtual server

A virtual server that accepts all traffic.

X

XenApp

A Citrix on-demand application delivery solution that enables any Windows application to be virtualized, centralized, and managed in the datacenter, and instantly delivered as a service to users anywhere on any device.

XenCenter

Management application for XenServer. You can use XenCenter to create, deploy, manage, and monitor virtual machines (VMs) from a Windows computer.

XenDesktop

A Citrix desktop virtualization and VDI solution that delivers a complete Windows desktop experience as an on-demand service to any user, anywhere.

XenServer

The Citrix open-source virtualization platform.

