This solution guide focuses on defining the process for configuring Kerberos Multi Domain Authentication for Exchange ActiveSync.
Table of Contents

Introduction 3
NetScaler features to be enabled 4
Solution Details 4
Deployment 7
Requirements 7
Windows Configuration 8
Workflow 9
NetScaler Configuration 10
Troubleshooting 13
Conclusion 14
Citrix NetScaler is a world-class application delivery controller (ADC) with the proven ability to load balance, accelerate, optimize and secure enterprise applications. A number of enterprises today use multiple domains to organize their employee credentials effectively. This complicates configurations for key enterprise services such as Microsoft Exchange based email.

**Introduction**
This guide focuses on defining a solution for enabling multi domain Kerberos SSO using Impersonation for Microsoft Exchange 2013 ActiveSync-based clients with Citrix NetScaler.

**Configuration Details**
The table below lists the minimum required software versions for this integration to work successfully. The integration process should also work with higher versions of the same.

<table>
<thead>
<tr>
<th>Product</th>
<th>Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>NetScaler</td>
<td>10.5 (Enterprise/Platinum License)</td>
</tr>
</tbody>
</table>
NetScaler features to be enabled
Before configuring Kerberos multi-domain authentication for ActiveSync, make sure that the following features are enabled on the NetScaler system.

Load balancing
NetScaler load balancing evenly distributes requests to backend servers. You can choose from a variety of algorithms (such as LEASTCONNECTION and ROUNDROBIN) to provide the most efficient load balancing logic for your application servers.

AAA-TM (Authentication, authorization and auditing (AAA) – Traffic Management (TM))
The NetScaler AAA feature set uses policies to control authentication, authorization, and auditing. These policies include definition and management of various authentication schemas. NetScaler supports a wide range of authentication protocols and a strong, policy-driven application firewall capability.

Solution details
AAA-TM with LDAP
To handle password changes by users, AAA-TM uses the method required by the protocol enabled on the authentication server. For most protocols, neither the user nor the administrator needs to do anything different than they would without AAA-TM. Even when an LDAP authentication server is in use, and that server is part of a distributed network of LDAP servers with a single designated domain administration server, password changes are usually handled seamlessly. When an authenticated client of an LDAP server changes his or her password, the client sends a credential-modify request to AAA-TM, which forwards it to the LDAP server. If the user’s LDAP server is also the domain administration server, it responds appropriately, and AAA-TM then performs the requested password change. Otherwise, the LDAP server sends AAA-TM an LDAP_REFERRAL response through the domain administration server. AAA-TM follows the referral to the indicated domain administration server, authenticates to that server, and performs the password change on that server.

When configuring AAA-TM with an LDAP authentication server, the system administrator must keep the following conditions and limitations in mind:

- AAA-TM assumes that the domain administration server in the referral accepts the same bind credentials as the original server.
- AAA-TM follows only LDAP referrals for password change operations. In other cases AAA-TM refuses to follow the referral.
- AAA-TM follows only one level of LDAP referrals. If the second LDAP server also returns a referral, AAA-TM refuses to follow the second referral.

The ADC supports auditing of all states and status information, so you can see the details of what each user did while logged on, in chronological order. To provide this information, the NetScaler appliance logs each event, as it occurs, either to a designated audit log file on the appliance or to a syslog server. Auditing requires configuring the appliance and any syslog server that you use.

**Kerberos**

Kerberos, a computer network authentication protocol, provides secure communication over the Internet. Designed primarily for client-server applications, it provides for mutual authentication by which the client and server can each ensure the other’s authenticity. Kerberos uses a trusted third party, referred to as Key Distribution Center (KDC). A KDC consists of an Authentication Server (AS), which authenticates a user, and a Ticket Granting Server (TGS).

Each entity on the network (client or server) has a secret key that is known only to itself and the KDC. The knowledge of this key implies authenticity of the entity. For communication between two entities on the network, the KDC generates a session key, referred to as the Kerberos ticket or service ticket. The client makes a request to the AS for credentials for a specific server. The client then receives a ticket, referred to as Ticket Granting Ticket (TGT). The client then contacts the TGS, using the TGT it received from the AS to prove its identity, and asks for a service. If the client is eligible for the service, the TGS issues a Kerberos ticket to the client. The client then contacts the server hosting the service (referred to as the service server), using the Kerberos ticket to prove that it is authorized to receive the service. The Kerberos ticket has a configurable lifetime. The client authenticates itself with the AS only once. If it contacts the physical server multiple times, it reuses the AS ticket.

The following figure shows the basic functioning of the Kerberos protocol.
When multi-realm, cross-domain Kerberos authentication needs to be configured, a trust relationship has to be established between the different realms. This trust can be direct, transitive or hierarchical.

Direct trust relationships are the basis of cross-authentication and are used to construct the other two types of relationships. In this type of trust relationship, the KDC of realm B has direct trust in the KDC of realm A, thus allowing the users of realm A to access realm B’s resources. From a practical point of view, a direct trust relationship is obtained by having the two involved KDCs share a key. The keys become two if bidirectional trust is required. To achieve this the concept of a remote Ticket Granting Ticket (RTGT) is introduced which, in the example of the two realms A and B, assumes the form krbtgt/B@A and is added to both the KDCs with the same key. This key is the secret that will guarantee the trust between the two realms. Obviously, to make it bi-directional (that is, A also trusts B), it is necessary to create the remote TGT in both KDCs (krbtgt/B@A and krbtgt/A@B), associating each of them with another secret key.

Kerberos 5 introduced the concept of transitive trust relationships to reduce complexity in the case of a large number of realms needing to be cross-authenticated; if realm A trusts realm B and realm B trusts realm C then A will automatically trust C. This relationship significantly reduces the number of keys required. Each time you create a new domain in a forest, a two-way, transitive trust relationship is automatically created between the new domain and its parent domain. If child domains are added to the new domain, the trust path flows upward through the domain hierarchy extending the initial trust path created between the new domain and its parent domain.

Hierarchical trust relationships are transitive trust relationships within a well-defined hierarchical domain setup. Kerberos can build authentication paths between adjacent realms automatically when the domains are configured appropriately.

**Solution Use Case**

This solution is relevant for enterprise use cases in which Exchange and other enterprise services are configured in various sub-realms/domains based on user role or other criteria, but user IDs exist in the parent realm/domain and must therefore be authenticated against the parent realm/domain.
**Deployment**

For this deployment, it is assumed that transitive trust has been established between the multiple domains between which Kerberos authentication has to be set up. In the following diagram, AAATM.com has transitive trust established with HR.AAATM.COM. The Exchange ActiveSync service is running on OWA2.HR.AAATM.COM.

**Multi-domain Deployment**

The following section is going to demonstrate multi-domain Kerberos impersonation of a user who belongs to AAATM.COM (parent domain) to a service that belongs to HR.AAATM.COM (child domain).

**Requirements**

Open the DMZ Firewall ports from the NetScaler SNIP address to internal servers (Exchange CAS, Domain Controllers, and DNS servers). Note that you must open UDP and TCP where specified.

<table>
<thead>
<tr>
<th>Port</th>
<th>Protocol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>53</td>
<td>UDP/TCP</td>
<td>DNS</td>
</tr>
<tr>
<td>88</td>
<td>UDP/TCP</td>
<td>Kerberos</td>
</tr>
<tr>
<td>123</td>
<td>UDP</td>
<td>NTP</td>
</tr>
<tr>
<td>135</td>
<td>TCP</td>
<td>RPC Endpoint Mapper</td>
</tr>
<tr>
<td>137</td>
<td>UDP</td>
<td>NetBIOS Name Service</td>
</tr>
<tr>
<td>139</td>
<td>TCP</td>
<td>NetBIOS Session (SMB)</td>
</tr>
<tr>
<td>389</td>
<td>UDP/TCP</td>
<td>LDAP</td>
</tr>
<tr>
<td>445</td>
<td>TCP</td>
<td>SMB over TCP</td>
</tr>
<tr>
<td>3268</td>
<td>TCP</td>
<td>Global Catalog Search</td>
</tr>
</tbody>
</table>
Prerequisites

- DNS should be configured to resolve all the domains.
- The ssoAttribute must be configured as (UserPrincipalName) UPN in the LDAP action, so that user’s realm can be properly retrieved from Active Directory/LDAP. Note that this is crucial for obtaining the user’s actual realm instead of the alternate UPN suffix or common domain.
- Transitive Trust must be established between the domains.

Windows Configuration

Kerberos configuration on HR.AAATM.COM Domain controller for Exchange server

Open the Active Directory Users and Computers prompt on the domain controller for the HR.AAATM.COM domain, then select the computer name representing the Exchange Server. Right click to view the properties as shown below.

As shown above, select the Delegation tab in the Properties window, then select Trust this computer for delegation to any service (Kerberos only). You should then configure the appropriate authentication settings on the ActiveSync virtual directory in Internet Information Server (IIS, the integrated Windows web server) on the Exchange server at HR.AAATM.COM.
IIS Web server settings for the ActiveSync virtual directory on the Exchange Server

As shown in the screenshot below, in the Authentication settings for the ActiveSync virtual directory, enable only Windows Authentication. Disable the other options.

**Workflow**

The diagram below describes the workflow for this solution.
Explanation of the Workflow
1. The client makes a request to either a Load Balancing (LB) or a Content Switching (CS) virtual server on a NetScaler appliance.
2. Client-side authentication begins with the handshake determined by the LDAP authentication policy configured on the virtual server. This document assumes that client side authentication is set to 401 Basic.
3. The User from the AAATM.com domain submits credentials to the NetScaler appliance.
4. The virtual server (vserver) verifies the client credential either directly, or through an external authentication server. The vserver extracts user’s UPN as part of authentication.
5. The virtual server selects a backend service and forwards the request to the backend exchange service, which responds with HTTP response 401 with WWW-Authentication: Negotiate.
6. The NetScaler appliance begins the Kerberos SSO process upon receiving the 401 response from the server if single sign on (SSO) has been enabled along with the configuration of a valid KCD Account. The KCD Account is picked as specified by the session/traffic policies configured on the appliance. In this case, if the user’s UPN is correctly extracted in step 4, you need only one KCDAccount.
7. The NetScaler Kerberos daemon (nskrb) communicates with the AAATM.com KDC (Key Distribution Center) to obtain Kerberos tickets. This is a multi-step process. The number of steps varies depending on the flow. The nskrb daemon sends a request for the TGT (ticket-granting ticket) for the user’s home domain, AAATM.COM.
8. The AAATM.com KDC returns a TGT for AAATM.COM.
9. The nskrb daemon sends a TGS (ticket granting server) request and the TGT for AAATM.COM to HR.AAATM.com.
10. The TGS service of HR.AAATM.com responds with a service ticket for the OWA2.HR.AAATM.COM Exchange ActiveSync service. HR.AAATM.COM recognizes a request for TGS with a foreign domain server and responds with a referral. The nskrb daemon performs necessary actions to obtain TGS for the exchange server.
11. The NetScaler appliance computes an HTTP Authorization header according to Kerberos protocol and sends it to the Exchange Server.
12. The backend service verifies the ticket in the HTTP request. If it is valid, the service grants access to the user and replies with the correct HTTP response.
13. The appliance receives the response and forwards the response to the client.

NetScaler Configuration
There are two ways of obtaining Kerberos tickets on a NetScaler appliance: Impersonation and Constrained Delegation. Impersonation can be done if the user’s Active Directory password is available on the appliance.

For this solution, impersonation can be used for Kerberos SSO to ActiveSync, because the front-end authentication scheme is basic (401-based).
Setting up NetScaler for User Impersonation

Create back-end servers and services

\texttt{add server <Exchange2013> OWA2.HR.AAATM.COM}

\texttt{Or}

\texttt{add server <Exchange2013> <IP Address>}

\texttt{add service Exchange2013_SRV Exchange2013 SSL 443}

Note: Here OWA2.HR.AAATM.com represents the backend server name, which should be replaced by the backend server’s name in your configuration.

Make sure the DNS resolution for <AAATM.com> and <Hr.AAATM.com> works as expected.

Create TM and Authentication Virtual Server

\texttt{add authentication vserver auth1 SSL <IP Address>443}

\texttt{set authentication vserver auth1 –authenticationdomain AAATM.COM}

\texttt{add lb vserver accesslb1 SSL <IP Address> <SSL Port>}

\texttt{bind lb vserver accesslb1 < Exchange2013_SRV>}

Enable Client Side Authentication (Form-based/401basic)

Enable the authentication method that you want on the TM virtual server.

\texttt{set lb vs <accesslb1> –authn401 –authnvsname <authentication Vserver name>}

where authnvsname is the name of the authentication vserver.

Create LDAP authentication policy

\texttt{add authentication ldapAction A.example -serverIP <Server IP> -ldapBase}

“\texttt{dc=A,dc=example,dc=com}” -\texttt{ldapBindDn administrator@AAATM.example.com -ldapBindDnPassword <Passwd> -encrypted -ldapLoginName samAccountName -groupAttrName memberOf}

-ssoNameAttribute UserPrincipalName

\texttt{add authentication ldapPolicy Ldapxmdpol ns_true Ldapxmd}

\texttt{bind Authentication vserver < auth1 >--policy Ldapxmdpolc}

Ensure that UserPrincipalName is configured as ssoAttrName. If the LDAP server does not return appropriate realm in User principal name, then please refer to the Troubleshooting section. Also do note that all the SSL vservers created (auth1 and accesslb1) will need a certificate bound to them for them to be accessible.
Create KcdAccount

KCDAccount is an entity consisting of the configuration necessary for delegation or impersonation. With delegation, a configured user gets tickets on behalf of an actual user of a service. Impersonation is used in cases such as the current one, in which the user’s password is available to the NetScaler appliance. With Impersonation, tickets are obtained by using the actual user’s credentials.

Create KcdAccount using Impersonation

add kcdaccount <OWA-Activesync> –realmStr HR.AAATM.COM

Create Traffic/Session Profile

You can provide Kcdaccount in a session action or traffic action. The NetScaler administrator can select from several rules. 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.

Note: When you specify kcdaccount along with SSO ON in a traffic/session profile, the NetScaler appliance completes a Kerberos SSO to the backend servers. If KcdAccount is set to “none”, the NetScaler appliance does NTLM SSO.

Settings in the Session Profile

add tm sessionAction Sess-OWA-SSO ON –kcdaccount OWA-Activesync
add tm sessionPolicy Sess-OWA-Pol TRUE  Sess-OWA
bind authentication vserver auth1 -policy Sess-OWA-Pol

Settings in the Traffic Profile

add tm trafficAction mytraffic-OWA -SSO ON –kcdAccount OWA-Activesync
add tm trafficPolicy mytraffic-A-Pol TRUE  mytraffic-OWA
bind lb vserver accesslb1 –policy mytraffic-A-Pol –priority 10
**Troubleshooting**

**LDAP Server fails to provide appropriate UserPrincipalName**

Some of the LDAP servers might not have UserPrincipalName reflecting user’s actual realm. If you require any other User realm to be used when requesting for TGS tickets, you can explicitly specify the userrealm parameter in the KCDAccount.

Example:

```
Add KCDAccount <Exchange> -userrealm <AAATM.COM> -realmstr <ServiceRealm>
```

Please note that if userRealm is specified in kcdAccount, all the users’s realms are modified if that kcdAccount is chosen. So, the administrator must configure one kcdAccount for every unique userRealm. One of these kcdAccounts must be chosen based on traffic policy.

**Multiple Services running on the same server and having an issue with SPN's**

There might be cases in which multiple services are running on the same machine, and managing the SPN might be challenging. In such a case you can get a ticket for HOST and use the ticket for multiple service running on that machine. To explicitly get a host ticket for the machine, you can specify the kcd account parameter “servicespn”.

Example

```
Add KCDAccount <Exchange> -realmstr <ServiceRealm> -serviceSPN HOST/<OWA2.HR.AATM.com>@ HR.AATM.com
```

**Multiple services run under same AppPoolCredentials**

Sometimes different backend servers can run with the same application pool credentials. Thus, servers like sharepoint.company.com or owa.company.com might all be using single SPN application.company.com even though their hostnames are different. Also, in such cases, a “ServiceSPN” configuration can be used in kcdAccount such that a Kerberos ticket is obtained for the appropriate entity.

```
Add KCDAccount <Exchange> -realmstr <ServiceRealm> -serviceSPN HTTP/application.company.com@company.com
```
Conclusion
This solution guide presents a specific solution for multi domain authentication for Exchange ActiveSync, with NetScaler, which is a common requirement across multiple enterprise and government organizations today.