Citrix NetScaler Gateway and Cisco ACI Integration Guide
Table of Contents
Introduction 3
NetScaler Gateway with XenApp and XenDesktop 3
Critical Requirements for Success with Application and Desktop Virtualization 3
Cisco ACI with the Citrix NetScaler ADC 4
Deploying Cisco ACI and NetScaler for Agile, Policy-driven Networks 4
The Unique Benefits of NetScaler with XenApp and XenDesktop 5
Overview 5
Cisco ACI 6
Citrix XenDesktop 7
XenDesktop and UCS 7
XenDesktop and NetScaler 8
Benefits of Citrix NetScaler and Cisco ACI Integration 8
Our Environment 9
Step-by-Step Guide for Integrating NetScaler Gateway and ACI 10
Creating the VPX on the SDX 10
Configuring the ACI Fabric to Connect to the SDX and VPX 16
Create the Logical VPX Device Inside of ACI 26
Create the Necessary Policy in ACI to Allow Communication 30
Configure the VPX through the APIC to Connect to the ACI Fabric 39
CS Service Graph 47
Access Gateway Service Graph 51
Test the Configuration 57
Summary 58
As businesses look to IT as a point of strategic differentiation, agility in the data center becomes more critical than ever. Fundamental to this change is the capability of IT to respond quickly to changing business requirements. Applications serve as the core of any business, but applications are only as agile as the infrastructure on which they run. With today's data center infrastructure, this rule can mean waiting weeks for an application change. Application agility, mobility, and rapid deployment require the data center infrastructure to dynamically respond to application needs as a result of changing business requirements.

Compared to traditional, distributed approaches for application and desktop deployment and management, virtualization solutions such as XenApp and XenDesktop promise significantly reduced operating costs, greater business agility, better data protection, and improved compliance with industry and corporate standards. Fully achieving these gains, however, depends on ensuring the availability, security, performance, and scalability of the associated infrastructure. As a result, leading IT organizations have developed the best practice of front-ending their application and desktop virtualization deployments with a dedicated solution that helps across all of these critical areas.

To achieve agility in deployment, secure access, and the best performance results for XenApp and XenDesktop, organizations need to combine the capabilities of Cisco Application Centric Infrastructure (ACI) with the Citrix NetScaler Application Delivery Controller (ADC) appliances. This integration reduces deployment complexity and better aligns applications to infrastructure automation using centralized policy-based management and ensures that users gain fast and secure access.

**NetScaler Gateway with XenApp and XenDesktop**

By combining an extensive set of capabilities for ensuring the accessibility of essential components, enhancing the user experience, and protecting associated data, applications, and infrastructure, NetScaler more than just preserves the benefits promised by XenApp, XenDesktop, and other application and desktop virtualization solutions. It maximizes them.

**Critical Requirements for Success with Application and Desktop Virtualization**

Enterprises that invest in full-featured application and desktop virtualization solutions such as XenApp and XenDesktop unlock the potential to:
• Sustainably reduce desktop ownership and operating costs
• Expand device compatibility by delivering Windows applications to both Windows and non-Windows devices
• Enable complete workplace flexibility, as well as workforce continuity, in the event of a disaster or disruption
• Improve security and meet compliance mandates by having desktops, applications, and data remain in the data center
• Increase business agility by rapidly and efficiently supporting strategic initiatives such as mergers and acquisitions, geographic expansion, and dynamic partnership arrangements

However, the extent to which your organization can achieve these and other available benefits will depend on how well your implementation addresses the following set of critical requirements.

**Cisco ACI with Citrix the NetScaler ADC**
The combined Cisco ACI and Citrix NetScaler solution provides a single point of management to define the network and L4 to L7 services requirements using policy-centric profiles, while elastically integrating them into the Cisco ACI network fabric. Cisco ACI and Citrix NetScaler ADC appliances combine to reduce deployment time to your physical and virtual application network services on a data center fabric. This combination offers freedom of deployment location, alignment with specific business applications, automated network configuration and services, plus ease of cleanup when the applications using services are decommissioned.

Cisco® Application Centric Infrastructure (ACI) is a scalable, resilient, and high-speed fabric that uses software-defined networking (SDN) concepts to automate network configuration. The physical fabric is created once; network services are then overlaid via centrally managed software-defined policies. ACI takes advantage of the full feature set of NetScaler App Delivery Controller (ADC) appliances, applying Layer 4 through Layer 7 network services to configure NetScaler to intelligently control traffic flows. ACI features a centralized point of control—the Application Policy Infrastructure Controller (APIC)—which administrators use to set up and manage the ACI infrastructure, including the insertion of NetScaler services. Using APIC, administrators configure network service policies in application-specific profiles. The combination of ACI and NetScaler technologies provides a flexible way to link network services to applications, thereby improving application security, increasing performance and optimizing service levels.

**Deploying Cisco ACI and NetScaler for Agile, Policy-driven Networks**
The ACI fabric connects networks to applications without the need to rewire or physically reconfigure network components as application requirements change. Built on a foundation of Cisco ASR 1000 routers and Cisco Nexus 9000 spine and leaf switches (Figure 1), ACI consolidates virtual and physical networks, supporting any-to-any routing and switching for optimal flexibility in cloud datacenters.
APIC provides a centralized interface to perform L2-L7 network management—including service insertion for the full range of L4-L7 NetScaler capabilities—rapidly building out production-ready configurations. Network administrators can use the APIC GUI to design and control networks or programmatically define configurations in APIC using XML or JSON.

NetScaler intelligently directs application traffic between the Cisco ACI fabric and the available infrastructure. Cisco recommends the use of NetScaler appliances when designing next-generation data center networks (or as a direct replacement to Cisco ACE products) because NetScaler is the only ADC in which the full feature set is exposed to the ACI fabric.

This comprehensive level of integration automates the process of aligning applications to infrastructure and reduces deployment complexity, increases agility, and speeds time-to-production.

**The Unique Benefits of NetScaler with XenApp and XenDesktop**

- SmartControl
- HDX insights
- Support for StoreFront
- Support for ThinWire+
- XenMobile and microVPN
- Support for any adaptive security policies (SmartAccess and SmartControl) with XA/XD
- SmartAccess
- Support for STA
- Support for FrameHawk
- Unified Gateway with ONE URL
- Visibility and monitoring of XA/XD traffic using HDX Insight
- Support for delivery of Thinwire+ and Framehawk

**Overview**

In this deployment guide we will be covering how to integrate Citrix NetScaler Access Gateway and Cisco ACI.

For this purpose we will have a working Citrix XenDesktop environment, a working Cisco UCS environment, and a working Cisco ACI environment.
We will describe the key features of these components and how they relate to one another.

Finally, we will cover the necessary steps to deploy the NetScaler and necessary changes we need to make to the ACI fabric for it to work.

**Cisco ACI**

The ACI fabric we are going to be using consists of two spines and two leaf switches. For those of you not familiar with ACI, it is Cisco’s next generation fabric. We can boil down ACI to these key concepts:

- Automation
- Policy Driven
- Highly Scalable
- Application Aware

ACI is built on the idea of promise theory. We tell the nodes of the fabric the end result we want to achieve but not how to achieve it.

We have a controller called the APIC, which is a cluster of at least three c220 appliances that will push policy onto the ACI fabric. There are several scripting tools for talking to the APIC.

The ACI fabric is deployed in a Clos network design that consist of spines and leaf switches. Spines only connect to leaf switches, leaf switches only connect to spines or endpoints. The Clos design makes the fabric more scalable—two endpoints are never more than one hop away from each other.
The policy model in ACI is based on a whitelist approach. We have endpoint groups (EPGs) that are a collection of endpoints we apply the same policies to. EPGs cannot reach each other unless they are explicitly allowed by a contract.

EPGs are brought together under an application profile. This is how the APIC has an end-to-end view for the applications inside of ACI, not just for servers of services running.

For this solution we will be using two EPGs: one for the clients and one for the servers. Each of the EPGs will have its own bridge domain (flooding boundary).


**Citrix XenDesktop**

The heart of our XenDesktop solution is the delivery controller. It’s responsible for managing end user access, contacting Active Directory to validate authentication requests, interacting with the database to retrieve the list of resources assigned to a user, and talking to the storefront to make those resources (apps and published desktops) available.

It balances requests, prepares the resources to be delivered to the end user, and load balances the connections to the storefront servers.

We hosted the Citrix Studio in the delivery controller. The Citrix Studio is the management console for the XenDesktop infrastructure.

The storefront server is the replacement of the web interface server. It authenticates end users to get access to the hosted desktops and apps in the XenDesktop infrastructure. Once the user credentials are validated, the authentication service handles subsequent interactions, so the user only needs to log on once. It uses centralized stores to deliver apps, desktop, and other resources to end users on any platform or endpoint.

To avoid a single point of failure and provide high availability, we deployed two delivery controllers and two storefront servers that we synchronized with each other. Citrix NetScaler is used to load balance and fail over between servers, so users have uninterrupted access to resources.

**XenDesktop and UCS**

The XenDesktop environment will be running on UCS, and both the ADC and UCS servers are powered by Intel Xeon processors. The fabric interconnects (FIs) are where all the configuration is done and provide the outside connectivity.

We created a service profile for each of the VMware ESXi hosts on which the virtual machines (VMs) for the Xen environment will be running.
The benefit of using UCS is that configuration is not applied to specific servers—it is applied to service profiles. In the event of a failure, the profile will get automatically applied to a different physical server.

We will use static pinnings for the VNICS we created on UCS and will match the VLAN range used on the physical domain on ACI.

You can find more information on UCS here:

**XenDesktop and NetScaler**

In our deployment we have a NetScaler VPX, with NetScaler Gateway enabled. It provides XenApp, XenDesktop, and XenMobile front end functionality, plus L4 – L7 traffic processing, such as load balancing and content switching among other features.

One of the core functionalities of NetScaler is the ability to load balance backend servers to provide high availability. NetScaler is validated on Intel Xeon processor-based servers.

In a XenDesktop environment, we load-balanced the storefront servers that end users rely on to access the published resources.

We also provide LDAP integration with Active Directory, endpoint analysis, HTTP to HTTPS redirection to provide secure remote access to our resources, and ICA Proxy, which gives external users access to resources located in the internal network via a single IP address and port, configured on the external interface of the firewall.


**Benefits of Citrix NetScaler and Cisco ACI Integration**

One of the key benefits of deploying NetScaler as the Application Delivery Controller (ADC) of choice for Cisco ACI is the seamless integration you get. Instead of having to pre-provision the ADC and then deploying services, we can use Service Graphs.

As we have already covered, one of the key concepts of ACI is that it is policy driven: EPGs can only talk to other EPGs if there is a specific policy allowing that traffic. That policy is called a contract.

A Service Graph is a means to specify not only the contract that is applied between two EPGs but also the L4 – L7 services they need to go through before the connection is allowed (SSL Offloading, LB, CS, etc.)
To be able to use a Service Graph, we need to first upload a device package into the APIC and create a L4 – L7 device.

The device package is the specified parameters for the device so that the APIC is able to push configurations to it (Push Scripts). The L4 – L7 is the logical construct on ACI that will represent the physical device.

When we deploy the service graph the configuration pushed to the device, all the necessary VLANs are dynamically provisioned, and the static bindings are created.

Our Environment

On the UCS side we created a port-channel that connects to the ACI leaf switches. We created the VLANs that are going to be used by the APIC and pinned them to the uplinks going to ACI.

CDP is going to be enabled between the FIs and the blades, and LLDP is going to be enabled northbound to the leaf switches. We also created the service profiles for the ESXi hosts, they have VNICs used for management and VNICs used to connect into ACI.

We need to create the VLAN pool on ACI that is going to match what we have on UCS. We will also create the VPC that connects to the FIs and the one that connects to the ESXi where the clients reside.

On ACI we will have LLDP enabled to talk to the FIs, CDP enabled to talk to the VMs, and LACP mac-pinning for the VMs connected to the VDS. (Fi's don't support VPC southbound.) For doing this we created the AEP and interface profiles.

A VMM domain will be created on ACI, and there we will need to specify the credentials for VCenter, the plan pool, and AEP we created will be applied here.

For the Xen environment we are using two delivery controllers, two storefront, one domain controller, and one published server. The domain controller is also the CA for the environment.

All the VMs have been joined to an AD domain.

We published an application on the store from the delivery controller.

This VM’s are connected to the same VDS, which was pushed from the APIC, and the port group is dynamically created when we associate the VMM domain to an EPG (shown later on this guide).

The 10.0.0.0/24 network is used for Servers EPG. 10.0.0.100 SNIP GW on the BD is the 10.0.0.1

The 192.168.101.0/24 network is used for Clients EPG. 192.168.101.100 VIP on the BD GW is the 192.168.101.1.
The 192.168.10.0/24 is the management network.

The VLANs used between ACI and UCS are 610-620 and 101.

The VLANs used between ACI and the SDX are 1300-1500, 610-620 and 101.

The domain used is citrixaci.lab.

**Step-by-Step Guide for Integrating NetScaler Gateway and ACI**

In this step-by-step guide we will be covering the specific steps needed for this integration:

- Create the VPX on the SDX
- Configure the ACI fabric to connect to the SDX and VPX
- Create the logical VPX device inside of ACI
- Create the necessary policy in ACI to allow communication
- Configure the VPX through the APIC to connect to the ACI Fabric
- Test our configuration

**Creating the VPX on the SDX**

The NetScaler we are using for this deployment is a VPX running version 11. This runs on an SDX, which is a physical device that allows us to create several VPX instances on top of it.

The first step we need to perform is create the VPX instance on the SDX. For this task, on the SDX, we need to go to Configuration/NetScaler/Instances and select Add.

On the next page we are required to enter all the relevant information for the VPX we are creating.

The name we use for this VPX is Citrix-ACI-VPX1, the management IP address is 192.168.10.149, the mask is 255.255.255.0, and the gateway is 192.168.10.1.
We choose the XVA file to be used by the VPX. The XVA file is the image the VPX is running. As we said before, for this guide we are using version 11.

For the feature license we are using Platinum, and the admin profile is the built-in ns_nsr progressBar profile.

Next, we assign the VPX the resources that the SDX will allocate to it. These numbers vary depending on the size of the deployment. In this step we are creating the credentials that will be used to administer this instance once it’s created.
Click **Add** under **Data Interfaces**.

Last we create the interfaces that the VPX is going to have and are used for data traffic. For this lab we are using **101,102,610-620,1300-1500**.

When creating this data interface we define the VLANs that are to be allowed for this instance. (As we define the plans here, they will be automatically created on the SDX.)

Since we are doing in band management, we select **L2VLAN**.
Click **Done**.

We now see a screen saying the instance is being provisioned, this might take two or three minutes.
Once this process finishes, we see that both the VM state and the instance state will show as up.

Next, we need to install the Root CA certificate and the Server Certificate. Go to Traffic Management > SSL > Certificates > Install.
Once we installed both certificates, we need to link the Server Certificate to the Root CA Certificate. We right-click on the Server Certificate, and click Link.
In this point of our deployment we have the VPX up and running and are able to start configuring it to connect to ACI.

**Configuring the ACI Fabric to Connect to the SDX and VPX**

As we said before, the SDX is going to be connected to our two leafs through a VPC. There are several steps that must be done before we can create it, the first of which is creating a physical domain on ACI. A physical domain is going to contain the ports and VLANs that are going to be used for a specific connection.

On the APIC GUI go to **Fabric/Access Policies/Physical and External Domains**, right-click **Physical Domains**, and select **Create Physical Domain**.
Now we create the Attachable Entity Profile (AEP). The AEP is a logical construct we create in order to define policies that are going to be later on applied to physical interfaces. The AEP will deploy the VLAN pools to the leaf switches we are using. (The VLANs on the ports will be applied when the EPG is applied.)

Under **Associated Attachable Entity Profile**, click **Create Attachable Entity Profile**.

Give the AEP a **Name** and click **Next**.
Click **Finish**. We will associate the interfaces later on.

To finish the physical domain configuration we are missing the VLANs that are going to be used.

Under **VLAN Pool**, select **Create VLAN Pool**.
The allocation mode is dynamic so it can be deployed as needed by the APIC when we create the EPGs and they need to reach the VPX. Click the + sign to add the VLAN block we are using. This VLAN range matches the VLANs we allowed on the data interfaces for our VPX.

Click **Ok**.
Click **Submit** to finish creating the VLAN pool.

Click **Submit** to finish the configuration of the physical domain.
We have now created the physical domain, the AEP, and the VLAN pool. In other words we have the containers and the VLAN definitions for the policy definitions we create for the interfaces we are using for our VPC.

The next step is creating the policy definitions for the VPC. For those unfamiliar with how VPC works on ACI, the main difference is that there is no peer link or peer keep alive between the leaf switches running the VPC. (As we stated before on ACI, a leaf will never be connected another leaf). This functionality is provided by the APIC.

Not shown here, we have already created a policy to disable CDP, a policy to enable LLDP, and a policy to set LACP to active.

You can go to Fabric/Access Policies/Policies and create those policies there.

On the Create VPC Interface Policy Group we define the policies we apply to the interfaces that are part of our VPC.

We choose here our CDP-Disable policy, LLDP-Enable, and LACP-Active. We also select the AEP we had created before for our physical domain. Click Submit.

As you might have noticed we defined the policies we want for our interfaces but we have not specified anywhere for those interfaces. This will be done on the next step, when we create the interface profile. Go to Fabric/Access Policies/Interface Policies, right-click Profile, and select Create Interface Profile.
Click the + sign to add the **Interface Selectors** (the physical interfaces that will take place on the VPC).

For **Interface ID** select the interface you will be using for the VPC. On the **Interface Policy** Group choose the one we created. Click **OK**.
Click Submit.

At this point, we are almost done with our VPC configuration; the only step we are missing is creating the switch profile.
The switch profile is used to define on which of all the LeafS we have on our ACI Fabric the Interface profile we created is going to get applied.

Go to Fabric/Access Policies/Switch Policies and right-click Create Switch Profile.

Under Switch Selectors choose the nodes you want to be part of the VPC (the nodes that are connected to the SDX).

Click Next.

Under Interface Selector Profiles select the one we created on the previous step.
Click **Finish**.

Once this step is finished the VPC should be up and running. To verify this go to **Fabric/Inventory/Pod/Leaf/Interfaces/VPC** and check if the VPC is up.

We now have connectivity between ACI and the VPX.

**Create the Logical VPX Device Inside of ACI**

In this section of the configuration guide, we create the L4-L7 device inside of ACI to represent the VPX we are using. We do this to configure the device from the APIC instead of locally.

From this L4-L7 device we configure the features and the modes for the NetScaler and the configuration parameters (VIPS, SNIPS, Vservers, etc.) are set from the service graph.

For the purpose of this guide, we are not using a service graph and will now show how to manually integrate the access gateway functionality.

The first task is uploading a device package for the specific device we want to create in ACI.

A device package is developed by the vendor of the L4-L7 device and contains the specifications for the APIC to talk to that device. (A script will push the configuration variables that the device understands.)

Go to **L4-L7 Services/Packages**, right-click **L4-L7 Service Device Packages**, and select **Import Device Package**.
Click **Browse** and select the device package (you downloaded this from the Citrix download page).

Click **Submit** when done. If we go to **L4-L7 Services/Inventory** we see the device package we just imported.

Once we have successfully imported the device package we are ready to create the L4-L7 device. When we create this L4-L7 device, we reference the device package we just created; that way the APIC knows the device it will be talking to. To create the L4-L7 device go to **Tenants/(Your Tenant)/L4-L7 Services**, right-click **L4-L7 Devices**, and select **Create L4-L7 Device**.

Next, we define our L4-L7 device.
Under **General**:
- **Name**: We are using Citrix-ACI-VPX1.
- **Device Package**: Select the device package we imported in the previous step.
- **Model**: In this case we are using a SDX Context.
- **Mode**: We are using single node.

Under **Connectivity**:
- **Physical Domain**: Choose the one we created that connects to the SDX.
- **APIC to Device Management Connectivity**: We are doing it In-Band.

Under **Credentials**:
- Specify the admin credentials you are using for the VPX Instance.

Under **Device 1**:
- **Management IP Address**: use the NSIP.
- **Connects To**: in this case we are using a VPC.
- **Physical Interfaces**: Specify a name, in our case it connects to the VPC we created and will be used both for providing and consuming contracts (explained later on).

Click **Next**.
Select **Cluster/All Parameters/NS Features** and enable the features for the NS; in this case we are using CS, LB, SSL, SSLVPN, RW, and RS.

Click **Finish**.

Go to **Tenants/(Your Tenant)/L4-L7 Services/L4-L7 Devices/(The device you just created)** and if the configuration was successful, under **Configuration State**, the **Device State** should be **Stable**.
You could also verify that the features you enabled from the APIC are now enabled on the VPX as well.

From the cli of the VPX do `show NS feature`.

At this point the VPX is ready to be configured from the APIC.

**Create the Necessary Policy in ACI to Allow Communication**

To create the policy configuration, we must first create a private network. You will often hear the private network called a context, and it is analogous to a VRF.

The private network is the first building block of our configuration, we then tie bridge domains to it, and finally tie the EPGs to the bridge domains.

You can think of the bridge domains as a flooding boundary and where we define the flooding behavior. We can have multiple EPGs and subnets on the bridge domain. For each subnet we define, we have a pervasive SVI, which is a gateway that will reside on all the leaf nodes that have an endpoint that belongs to an EPG that is bound to that bridge domain.

An EPG is a collection of endpoints bound to the same group that share the same policy and are bound to the same bridge domain. The EPGs are where we apply the contracts that are the policy we are applying.

Go to **Tenants/Networking**, right-click **Private Networks**, and select **Create Private Network**.
Give the Private Network a name. Don’t select Create a Bridge Domain, we will do this later.

Click Finish.

We now create two bridge domains (one for the client EPG and one for the server EPG). Go to Tenants/Networking, right-click Bridge Domains, and select Create Bridge Domain.
Give the bridge domain a **Name**, and for the **Network**, choose the **private network** we created in the previous step.

Click **Finish**.

We will now create a pervasive SVI for the bridge domain. (This is a default gateway that is going to exist in all the leaf switches—you could have multiple subnets per bridge domain.)

Go to Tenants/Networking/Bridge Domains/“your Server Bridge Domain”/Subnets.
The scope is going to define if we want the subnet to be published to the outside world if we are using dynamic routing protocols.
Click **Submit**. Repeat the steps for the Client Bridge Domain, using the appropriate subnet.

We will now create the application profile, which is the logical representation of the application. You can think of the EPGs as the services and components for that. The benefit of using an application profile is that we can ensure the application is behaving end to end.

Go to **Tenants/(Your Tenant)**, right-click **Application Profiles**, and select **Create Application Profile**.

Give the application profile a **name** and click **Submit**.

Now we can create the EPGs.
Go to Tenants/Application Profiles/(My App), right-click Application EPGs, and select Create Application EPG. Create one EPG for the servers and one for the clients.

Give the EPG a name for Bridge Domain and select the bridge domain we created before.

Click Finish.

We must define who is going to belong to that EPG. For this we use the VMM Domain. (The VMM domain is the connection from the APIC to VCenter, so the APIC is able to see the VMs running on VCenter, and push the port Groups from the APIC).

Right-click Domains and select Add VMM Domain Association.
For the **VMM Domain Profile** select the **VMM domain** you are using to connect to VCenter.

Click **Submit**.

We need to add to this EPG the physical domain that connects to the VPX. Right-click Domains and select **Add Physical Domain Association**.
Select the **physical domain** we created to connect to the SDX.

Click **Submit**.

Repeat these steps to create a client EPG. (Add both the VMM domain and physical domain.)

If you were to test the configuration up to this point you would see it is not yet working. A ping from VPX to Pervasive SVI (10.0.0.1) on the leaf fails.
A ping from the VPX to one of the XenDesktop Delivery Controller also fails (10.0.0.8)

**Configure the VPX through the APIC to Connect to the ACI Fabric**

We have already configured the APIC to talk to the VPX (when we created the L4-L7 device), now we create the Service Graph in order to be able to push the configuration to the VPX. We will be creating two Graphs: one for CS and one for access gateway.

Go to Tenants/L4-L7 Services, right-click L4-L7 Service Graph Templates, and click Create a L4-L7 Service Graph Template.

Select One Node and the deployment type is Single Node - ADC in Two-Arm Mode.

For device function choose Content Switching and for profile CS-SSL-LB-Service-Profile.
Click **Submit**.

Repeat this step for the other service graph. Use **SSLVPN** for the device function and **SSLVPNProfile** for the profile and click **Submit**.
The next step is to create the contracts that these service graphs will use. For the purpose of this guide we are using a default filter (which means all traffic is allowed).

Go to Tenants/Security Policies, right-click Contracts, and select Create Contract.

Give the contract a Name, and leave the Scope as Private Network. Click the + sign next to Subjects.
Give the subject a **Name** and click the + sign next to **Filters**

From the drop down menu select **common/default** and click **Update**.

Click **OK** on the **Create Contract Subject** window and click **Submit** on the **Create Contract** window.
Repeat these steps to create a second contract for the second service graph.

Under Security Policies/Contracts, expand the contract you create and select the Subject. On the Service Graph drop down menu chose the Service Graph that corresponds to the contract. (Repeat this step for the second contract you created and choose the second service graph.)

Click Submit.

We now need to create the device selection policies for the service graphs previously created. Go to Tenants/L4-L7 Services, right-click Device Selection Policies, and click Create Logical Device Context.
Choose the **Contract**, **Service Graph**, and **Device** we already created. For the **Node Name** select **ADC** from the drop down menu. Click the + sign next to **Logical Interface Contexts**.

From the drop down menu select **External under Logical Interface**. Click **Update**. Repeat this step to create an **Internal Logical Interface** as well.
Click **Submit**.

Repeat these steps to create a second Logical Device Context. Choose the second contract and service graph.

To apply this service graph we need to apply the contracts we created to our provider EPG (the server EPG we created before). Go to **Tenants/Application Profiles/(My App)/(My Provider EPG)**, right-click **Contracts**, and select **Add Provided Contract**.

Select the **Contract** from the drop down menu and click **Submit**.

Repeat these steps for the second contract.
The last step is to push the parameters for the service graph to the APIC. We are using XML to push the configuration, and Postman is the tool we are using.
CS Service Graph

```xml
<fvAEPg dn="uni/tn-Citrix-ACI/ap-Citrix-ACI-MyApp/epg-Citrix-ACI-Win-EPG" name="Citrix-ACI-Win-EPG"><vnsFolderInst ctrctNameOrLbl="Citrix-ACI-CS-Ctrct" 
graphNameOrLbl="Citrix-NetScaler-CS-Graph" nodeNameOrLbl="ADC" key="csvserver" 
name="csvserver_0">

  <vnsParamInst name="name" key="name" value="citrixaci_ns2"/>
  <vnsParamInst name="servicetype" key="servicetype" value="SSL"/>
  <vnsParamInst name="ipv46" key="ipv46" value="192.168.101.101"/>
  <vnsParamInst name="port" key="port" value="443"/>
  <vnsParamInst name="clttimeout" key="clttimeout" value="180"/>

</vnsFolderInst>

<vnsFolderInst ctrctNameOrLbl="Citrix-ACI-CS-Ctrct" graphNameOrLbl="Citrix-NetScaler-CS-Graph" nodeNameOrLbl="ADC" key="lbmonitor" name="lbmonitor_1">

  <vnsParamInst name="monitorname" key="monitorname" value="XD_SF_MON_192.168.101.101"/>
  <vnsParamInst name="type" key="type" value="STOREFRONT"/>
  <vnsParamInst name="scriptname" key="scriptname" value="nssf.pl"/>
  <vnsParamInst name="dispatcherip" key="dispatcherip" value="127.0.0.1"/>
  <vnsParamInst name="dispatcherport" key="dispatcherport" value="3013"/>
  <vnsParamInst name="secure" key="secure" value="YES"/>
  <vnsParamInst name="storename" key="storename" value="unauthstore"/>

</vnsFolderInst>

<vnsFolderInst ctrctNameOrLbl="Citrix-ACI-CS-Ctrct" graphNameOrLbl="Citrix-NetScaler-CS-Graph" nodeNameOrLbl="ADC" key="lbvserver" name="lbvserver_2">

  <vnsParamInst name="name" key="name" value="192.168.101.100http_redirect"/>
  <vnsParamInst name="servicetype" key="servicetype" value="HTTP"/>
  <vnsParamInst name="ipv46" key="ipv46" value="192.168.101.100"/>
  <vnsParamInst name="port" key="port" value="80"/>
  <vnsParamInst name="persistencetype" key="persistencetype" value="NONE"/>
  <vnsParamInst name="redirurl" key="redirurl" value="https://192.168.101.100"/>
  <vnsParamInst name="clttimeout" key="clttimeout" value="180"/>

</vnsFolderInst>

<vnsFolderInst ctrctNameOrLbl="Citrix-ACI-CS-Ctrct" graphNameOrLbl="Citrix-NetScaler-CS-Graph" nodeNameOrLbl="ADC" key="servicegroup" name="servicegroup_3">

  <vnsParamInst name="servicegroupname" key="servicegroupname" value="XD_WISF_SVCGRP_192.168.101.101"/>

</vnsFolderInst>
```
</vnsFolderInst>

<vnsFolderInst ctcterNameOrLbl="Citrix-ACI-CS-Ctrct" graphNameOrLbl="Citrix-NetScaler-CS-Graph" nodeNameOrLbl="ADC" key="servicegroup_servicegroupmember_binding" name="servicegroup_8">
    <vnsParamInst name="ip" key="ip" value="10.0.0.12"/>
    <vnsParamInst name="port" key="port" value="443"/>
</vnsFolderInst>
</vnsFolderInst>
<vnsFolderInst ctcterNameOrLbl="Citrix-ACI-CS-Ctrct" graphNameOrLbl="Citrix-NetScaler-CS-Graph" nodeNameOrLbl="ADC" key="servicegroup_servicegroupmember_binding" name="servicegroup_9">
    <vnsParamInst name="ip" key="ip" value="10.0.0.11"/>
</vnsFolderInst>
</vnsFolderInst>
<vnsFolderInst ctcterNameOrLbl="Citrix-ACI-CS-Ctrct" graphNameOrLbl="Citrix-NetScaler-CS-Graph" nodeNameOrLbl="ADC" key="lbmonitor_binding" name="servicegroup_7">
    <vnsCfgRelInst key="monitor_name" name="monitor_name" targetName="lbmonitor_1"/>
</vnsFolderInst>
</vnsFolderInst>
<vnsFolderInst ctcterNameOrLbl="Citrix-ACI-CS-Ctrct" graphNameOrLbl="Citrix-NetScaler-CS-Graph" nodeNameOrLbl="ADC" key="sslcertkey" name="sslcertkey_4">
    <vnsParamInst name="certkey" key="certkey" value="CertKeyPair"/>
    <vnsParamInst name="cert" key="cert" value="citrixacicert.cer"/>
    <vnsParamInst name="key" key="key" value="citrixaci.key"/>
    <vnsParamInst name="passplain" key="passplain" value="1234Qwer"/>
</vnsFolderInst>
</vnsFolderInst>
<vnsFolderInst ctcterNameOrLbl="Citrix-ACI-CS-Ctrct" graphNameOrLbl="Citrix-NetScaler-CS-Graph" nodeNameOrLbl="ADC" key="sslvserver" name="sslvserver_5">
    <vnsParamInst name="vservername" key="vservername" value="citrixaci_ns2"/>
</vnsFolderInst>
<vnsFolderInst ctcterNameOrLbl="Citrix-ACI-CS-Ctrct" graphNameOrLbl="Citrix-NetScaler-CS-Graph" nodeNameOrLbl="ADC" key="sslvserver_sslcertkey_binding" name="sslvserver_10">
    <vnsCfgRelInst key="certkeyname" name="certkeyname" targetName="sslcertkey_4"/>
    <vnsParamInst name="ca" key="ca" value="false"/>
    <vnsParamInst name="skipcaname" key="skipcaname" value="false"/>
    <vnsParamInst name="snicert" key="snicert" value="false"/>
Deployment Guide

Citrix NetScaler Gateway and Cisco ACI Integration Guide

</vnsFolderInst>

<vnsFolderInst ctrctNameOrLbl="Citrix-ACI-CS-Ctrct" graphNameOrLbl="Citrix-NetScaler-CS-Graph" nodeNameOrLbl="ADC" key="mFngservicegroup" name="vnsAbs_servicegroup_3">
<vnsCfgRelInst name="servicegroup_key" key="servicegroup_key" targetName="servicegroup_3"/>
</vnsFolderInst>

<vnsFolderInst ctrctNameOrLbl="Citrix-ACI-CS-Ctrct" graphNameOrLbl="Citrix-NetScaler-CS-Graph" nodeNameOrLbl="ADC" key="mFnglbvserver" name="vnsAbs_lbvserver_2">
<vnsCfgRelInst name="lbvserver_key" key="lbvserver_key" targetName="lbvserver_2"/>
</vnsFolderInst>

<vnsFolderInst ctrctNameOrLbl="Citrix-ACI-CS-Ctrct" graphNameOrLbl="Citrix-NetScaler-CS-Graph" nodeNameOrLbl="ADC" key="mFnglbmonitor" name="vnsAbs_lbmonitor_1">
<vnsCfgRelInst name="lbmonitor_key" key="lbmonitor_key" targetName="lbmonitor_1"/>
</vnsFolderInst>

<vnsFolderInst ctrctNameOrLbl="Citrix-ACI-CS-Ctrct" graphNameOrLbl="Citrix-NetScaler-CS-Graph" nodeNameOrLbl="ADC" key="mFngsslcertkey" name="vnsAbs_sslcertkey_4">
<vnsCfgRelInst name="sslcertkey_key" key="sslcertkey_key" targetName="sslcertkey_4"/>
</vnsFolderInst>

<vnsFolderInst ctrctNameOrLbl="Citrix-ACI-CS-Ctrct" graphNameOrLbl="Citrix-NetScaler-CS-Graph" nodeNameOrLbl="ADC" key="mFngsslvserver" name="vnsAbs_sslvserver_5">
<vnsCfgRelInst name="sslvserver_key" key="sslvserver_key" targetName="sslvserver_5"/>
</vnsFolderInst>

<vnsFolderInst ctrctNameOrLbl="Citrix-ACI-CS-Ctrct" graphNameOrLbl="Citrix-NetScaler-CS-Graph" nodeNameOrLbl="ADC" key="mFngsslvserver" name="vnsAbs_sslvserver_6">
<vnsCfgRelInst name="sslvserver_key" key="sslvserver_key" targetName="sslvserver_6"/>
</vnsFolderInst>

<vnsFolderInst ctrctNameOrLbl="Citrix-ACI-CS-Ctrct" graphNameOrLbl="Citrix-NetScaler-CS-Graph" nodeNameOrLbl="ADC" key="internal_network" name="internal_network">
<vnsCfgRelInst name="internal_network_key" key="internal_network_key" targetName="Network/snip1"/>
</vnsFolderInst>
Access Gateway Service Graph

<vnsFolderInst ctrctNameOrLbl="Citrix-ACI-Gateway-Ctrct" graphNameOrLbl="Citrix-NetScaler-Gateway-Graph" nodeNameOrLbl="ADC" key="authenticationldapaction" name="authnictionldapaction_2">
    <vnsParamInst name="name" key="name" value="Citrix-ACI-LDAP"/>
    <vnsParamInst name="serverip" key="serverip" value="10.0.0.10"/>
    <vnsParamInst name="ldapbase" key="ldapbase" value="DC=Citrixaci,DC=Lab"/>
    <vnsParamInst name="ldapbinddn" key="ldapbinddn" value="Administrator@citrixaci.lab"/>
    <vnsParamInst name="ldapbinddnpassword" key="ldapbinddnpassword" value="1234Qwer"/>
    <vnsParamInst name="ldaploginname" key="ldaploginname" value="sAMAccountName"/>
    <vnsParamInst name="groupattrname" key="groupattrname" value="memberOf"/>
    <vnsParamInst name="subattributename" key="subattributename" value="cn"/>
    <vnsParamInst name="ssonameattribute" key="ssonameattribute" value="cn"/>
</vnsFolderInst>

<vnsFolderInst ctrctNameOrLbl="Citrix-ACI-Gateway-Ctrct" graphNameOrLbl="Citrix-NetScaler-Gateway-Graph" nodeNameOrLbl="ADC" key="vpnsessionaction" name="vpnsessionaction_4">
    <vnsParamInst name="name" key="name" value="AC_WB_192.168.101.101"/>
</vnsFolderInst>
<vnsParamInst name="transparentinterception" key="transparentinterception" value="ON"/>
<vnsParamInst name="defaultauthorizationaction" key="defaultauthorizationaction" value="ALLOW"/>
<vnsParamInst name="sso" key="sso" value="ON"/>
<vnsParamInst name="icaproxy" key="icaproxy" value="ON"/>
<vnsParamInst name="wihome" key="wihome" value="https://10.0.0.11/citrix/unauthstoreweb"/>
<vnsParamInst name="clientchoices" key="clientchoices" value="ON"/>
<vnsParamInst name="ntdomain" key="ntdomain" value="citrixaci.lab"/>
<vnsParamInst name="clientlessvpnmode" key="clientlessvpnmode" value="OFF"/>
</vnsFolderInst>

<vnsFolderInst ctrailNameOrLbl="Citrix-ACI-Gateway-Ctrct" graphNameOrLbl="Citrix-NetScaler-Gateway-Graph" nodeNameOrLbl="ADC" key="vpnsessionaction" name="vpnsessionaction_5">
<vnsParamInst name="name" key="name" value="AC_OS_192.168.101.101"/>
<vnsParamInst name="transparentinterception" key="transparentinterception" value="OFF"/>
<vnsParamInst name="defaultauthorizationaction" key="defaultauthorizationaction" value="ALLOW"/>
<vnsParamInst name="sso" key="sso" value="ON"/>
<vnsParamInst name="icaproxy" key="icaproxy" value="ON"/>
<vnsParamInst name="wihome" key="wihome" value="https://10.0.0.11/citrix/unauthstoreweb"/>
<vnsParamInst name="clientchoices" key="clientchoices" value="OFF"/>
<vnsParamInst name="ntdomain" key="ntdomain" value="citrixaci.lab"/>
<vnsParamInst name="clientlessvpnmode" key="clientlessvpnmode" value="OFF"/>
<vnsParamInst name="splittunnel" key="splittunnel" value="OFF"/>
<vnsParamInst name="storefronturl" key="storefronturl" value="https://10.0.0.11"/>
</vnsFolderInst>

<vnsFolderInst ctrailNameOrLbl="Citrix-ACI-Gateway-Ctrct" graphNameOrLbl="Citrix-NetScaler-Gateway-Graph" nodeNameOrLbl="ADC" key="vpnsessionpolicy" name="vpnsessionpolicy_5">
<vnsParamInst name="name" key="name" value="192.168.100.100_443_POL"/>
<vnsParamInst name="rule" key="rule" value="NS_TRUE"/>
<vnsCfgrRelInst name="action" key="action" targetName="vpnsessionaction_3"/>
</vnsFolderInst>

<vnsFolderInst ctrailNameOrLbl="Citrix-ACI-Gateway-Ctrct" graphNameOrLbl="Citrix-NetScaler-Gateway-Graph" nodeNameOrLbl="ADC" key="vpnsessionpolicy" name="vpnsessionpolicy_6"/>
<vnsParamInst name="name" key="name" value="PL_OS_192.168.101.101"/>
<vnsParamInst name="rule" key="rule" value="REQ.HTTP.HEADER User-Agent CONTAINS CitrixReceiver'/">
<vnsCfgRelInst name="action" key="action" targetName="vpnsessionaction_5"/>
</vnsFolderInst>

<vnsFolderInst

<vnsFolderInst ctrctNameOrLbl="Citrix-ACI-Gateway-Ctrct" graphNameOrLbl="Citrix-NetScaler-Gateway-Graph" nodeNameOrLbl="ADC" key="vpnsessionpolicy" name="vpnsessionpolicy_7">
<vnsParamInst name="name" key="name" value="PL_WB_192.168.101.101"/>
<vnsParamInst name="rule" key="rule" value="REQ.HTTP.HEADER User-Agent NOTCONTAINS CitrixReceiver &amp;amp; REQ.HTTP.HEADER Referer EXISTS" />
<vnsCfgRelInst name="action" key="action" targetName="vpnsessionaction_4"/>
</vnsFolderInst>

<vnsFolderInst

<vnsFolderInst ctrctNameOrLbl="Citrix-ACI-Gateway-Ctrct" graphNameOrLbl="Citrix-NetScaler-Gateway-Graph" nodeNameOrLbl="ADC" key="authenticationldappolicy" name="authPol">
<vnsParamInst name="name" key="name" value="ad.citrixaci.lab_LDAP_pol"/>
<vnsParamInst name="rule" key="rule" value="NS_TRUE"/>
<vnsCfgRelInst name="reqaction" key="reqaction" targetName="authenticationldapaction_2"/>
</vnsFolderInst>

<vnsFolderInst

<vnsFolderInst ctrctNameOrLbl="Citrix-ACI-Gateway-Ctrct" graphNameOrLbl="Citrix-NetScaler-Gateway-Graph" nodeNameOrLbl="ADC" key="vpnvserver" name="vpnvserver_7">
<vnsParamInst name="name" key="name" value="citrixaci_ns" />
<vnsParamInst name="servicetype" key="servicetype" value="SSL"/>
<vnsParamInst name="ipv46" key="ipv46" value="192.168.101.100"/>
<vnsParamInst name="port" key="port" value="443"/>
<vnsParamInst name="listenpolicy" key="listenpolicy" value="NONE"/>
<vnsFolderInst ctrctNameOrLbl="Citrix-ACI-Gateway-Ctrct" graphNameOrLbl="Citrix-NetScaler-Gateway-Graph" nodeNameOrLbl="ADC" key="vpnvserver_vpnvsessionpolicy_binding" name="vpnvserver_8">
<vnsCfgRelInst key="policy" name="policy" targetName="Policy/vpnsessionpolicy_5"/>
<vnsParamInst name="priority" key="priority" value="62000"/>
</vnsFolderInst>
</vnsFolderInst>

<vnsFolderInst cardinality="unspecified" ctrctNameOrLbl="Citrix-ACI-Gateway-Ctrct" devCtxLbl="" graphNameOrLbl="Citrix-NetScaler-Gateway-Graph"
key="vpnvserver_vpnsessionpolicy_binding" locked="no" name="vpnvserver_82" 
odeNameOrLbl="ADC" scopedBy="epg"> 
  <vnsCfgRelInst cardinality="unspecified" key="policy" locked="no" mandatory="no" name="policy" targetName="Policy/vpnsessionpolicy_6" /> 
  <vnsParamInst cardinality="unspecified" key="priority" locked="no" mandatory="no" name="priority" validation="" value="110" />
</vnsFolderInst>

<vnsFolderInst cardinality="unspecified" ctrctNameOrLbl="Citrix-ACI-Gateway-Ctrct" devCtxLbl="" graphNameOrLbl="Citrix-NetScaler-Gateway-Graph" key="vpnvserver_vpnsessionpolicy_binding" locked="no" name="vpnvserver_vpnsessionpolicy_binding" locked="no" name="vpnvserver_vpnsessionpolicy_binding" locked="no" name="vpnvserver_vpnsessionpolicy_binding" locked="no" nodeOrLbl="ADC" scopedBy="epg"> 
  <vnsCfgRelInst cardinality="unspecified" key="policy" locked="no" mandatory="no" name="policy" targetName="Policy/vpnsessionpolicy_7" /> 
  <vnsParamInst cardinality="unspecified" key="priority" locked="no" mandatory="no" name="priority" validation="" value="100" />
</vnsFolderInst>

<vnsFolderInst cardinality="unspecified" ctrctNameOrLbl="Citrix-ACI-Gateway-Ctrct" devCtxLbl="" graphNameOrLbl="Citrix-NetScaler-Gateway-Graph" key="vpnvserver_authenticationldappolicy_binding" locked="no" name="vpnvserver_authenticationldappolicy_binding" locked="no" name="vpnvserver_authenticationldappolicy_binding" locked="no" nodeOrLbl="ADC" scopedBy="epg"> 
  <vnsCfgRelInst cardinality="unspecified" key="policy" locked="no" mandatory="no" name="policy" targetName="Policy/authPol" /> 
  <vnsParamInst cardinality="unspecified" key="priority" locked="no" mandatory="no" name="priority" validation="" value="100" />
</vnsFolderInst>

<vnsFolderInst cardinality="unspecified" ctrctNameOrLbl="Citrix-ACI-Gateway-Ctrct" devCtxLbl="" graphNameOrLbl="Citrix-NetScaler-Gateway-Graph" key="vpnvserver_staserver_binding" locked="no" name="vpnvserver_staserver_binding" locked="no" name="vpnvserver_staserver_binding" locked="no" nodeOrLbl="ADC" scopedBy="epg"> 
  <vnsParamInst cardinality="unspecified" key="staserver" locked="no" mandatory="no" name="staserver" validation="" value="http://10.0.0.8" />
</vnsFolderInst>

<vnsFolderInst cardinality="unspecified" ctrctNameOrLbl="Citrix-ACI-Gateway-Ctrct" devCtxLbl="" graphNameOrLbl="Citrix-NetScaler-Gateway-Graph" key="vpnvserver_staserver_binding2" locked="no" name="vpnvserver_staserver_binding2" locked="no" nodeOrLbl="ADC" scopedBy="epg"> 
  <vnsParamInst cardinality="unspecified" key="staserver" locked="no" mandatory="no" name="staserver" validation="" value="http://10.0.0.8" />
</vnsFolderInst>
graphNameOrLbl="Citrix-NetScaler-Gateway-Graph" nodeNameOrLbl="ADC"
key="mFCngPolicy" name="vnsAbs_Policy_6">
<vnsCfgRelInst name="Policy_key" key="Policy_key" targetName="Policy/Policy_6"/>

</vnsFolderInst>

<vnsFolderInst ctrctNameOrLbl="Citrix-ACI-Gateway-Ctrct" graphNameOrLbl="Citrix-NetScaler-Gateway-Graph" nodeNameOrLbl="ADC" key="mFCngPolicy"
name="vnsAbs_Policy_8">
<vnsCfgRelInst name="Policy_key" key="Policy_key" targetName="Policy/authPol"/>

</vnsFolderInst>

<vnsFolderInst ctrctNameOrLbl="Citrix-ACI-Gateway-Ctrct" graphNameOrLbl="Citrix-NetScaler-Gateway-Graph" nodeNameOrLbl="ADC" key="mFCngPolicy"
name="vnsAbs_Policy_7">
<vnsCfgRelInst name="Policy_key" key="Policy_key" targetName="Policy/vpnsessionpolicy_7"/>

</vnsFolderInst>

<vnsFolderInst ctrctNameOrLbl="Citrix-ACI-Gateway-Ctrct" graphNameOrLbl="Citrix-NetScaler-Gateway-Graph" nodeNameOrLbl="ADC" key="mFCngvpnvserver"
name="vnsAbs_vpnvserver_7">
<vnsCfgRelInst name="vpnvserver_key" key="vpnvserver_key" targetName="vpnvserver_7"/>

</vnsFolderInst>

<vnsFolderInst ctrctNameOrLbl="Citrix-ACI-Gateway-Ctrct" graphNameOrLbl="Citrix-NetScaler-Gateway-Graph" nodeNameOrLbl="ADC" key="internal_network"
name="internal_network">
<vnsCfgRelInst name="internal_network_key" key="internal_network_key"
targetName="Network/snip1"/>

</vnsFolderInst>

<vnsFolderInst ctrctNameOrLbl="Citrix-ACI-Gateway-Ctrct" graphNameOrLbl="Citrix-NetScaler-Gateway-Graph" nodeNameOrLbl="ADC" key="external_network"
name="external_network">
<vnsCfgRelInst name="external_network_key" key="external_network_key"
targetName="Network/snip2"/>

</vnsFolderInst>

</fvAEPg>
**Test the Configuration**

Log in to any of the VMs that belong to the client EPG. In a web browser point to the VIP of the Access Gateway VPN Vserver. You should reach the login page of the NetScaler Gateway.

Once you login you should reach the page to choose how you want to connect.

We can now reach the available resources.
Summary
Cisco ACI is the foundation of an application-based data center. Citrix NetScaler ADCs deliver application insight to the network. Together, Cisco ACI and ADC-enabled applications can dynamically scale and migrate throughout data centers on demand and with an automated approach based on application-specific policies.

About Citrix
Citrix (NASDAQ:CTXS) is leading the transition to software-defining the workplace, uniting virtualization, mobility management, networking and SaaS solutions to enable new ways for businesses and people to work better. Citrix solutions power business mobility through secure, mobile workspaces that provide people with instant access to apps, desktops, data and communications on any device, over any network and cloud. With annual revenue in 2014 of $3.14 billion, Citrix solutions are in use at more than 330,000 organizations and by over 100 million users globally. Learn more at www.citrix.com.

Copyright © 2016 Citrix Systems, Inc. All rights reserved. Citrix, NetScaler, NetScaler Gateway, XenApp and XenDesktop are trademarks of Citrix Systems, Inc. and/or one of its subsidiaries, and may be registered in the U.S. and other countries. Other product and company names mentioned herein may be trademarks of their respective companies.