# Cisco APIC Layer 4 to Layer 7 Service Graph Deployment Guide, Release 1.2(2g)

**First Published:** April 20, 2016

**Americas Headquarters**
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

# CONTENTS

# Preface

This preface includes the following sections:



## Audience

This guide is intended primarily for data center administrators with responsibilities and expertise in one or more of the following:

- Virtual machine installation and administration
- Layer 4 to Layer 7 Services installation and administration
- Switch and network administration

## Document Conventions

Command descriptions use the following conventions:

| Convention | Description |
|---|---|
| **bold** | Bold text indicates the commands and keywords that you enter literally as shown. |
| *Italic* | Italic text indicates arguments for which the user supplies the values. |
| [x] | Square brackets enclose an optional element (keyword or argument). |

| Convention | Description |
|---|---|
| [x \| y] | Square brackets enclosing keywords or arguments separated by a vertical bar indicate an optional choice. |
| {x \| y} | Braces enclosing keywords or arguments separated by a vertical bar indicate a required choice. |
| [x {y \| z}] | Nested set of square brackets or braces indicate optional or required choices within optional or required elements. Braces and a vertical bar within square brackets indicate a required choice within an optional element. |
| `variable` | Indicates a variable for which you supply values, in context where italics cannot be used. |
| string | A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks. |

Examples use the following conventions:

| Convention | Description |
|---|---|
| `screen font` | Terminal sessions and information the switch displays are in screen font. |
| **`boldface screen font`** | Information you must enter is in boldface screen font. |
| *italic screen font* | Arguments for which you supply values are in italic screen font. |
| < > | Nonprinting characters, such as passwords, are in angle brackets. |
| [ ] | Default responses to system prompts are in square brackets. |
| !, # | An exclamation point (!) or a pound sign (#) at the beginning of a line of code indicates a comment line. |

This document uses the following conventions:

**Note** Means *reader take note*. Notes contain helpful suggestions or references to material not covered in the manual.

**Caution** Means *reader be careful*. In this situation, you might do something that could result in equipment damage or loss of data.

**Warning** IMPORTANT SAFETY INSTRUCTIONS

This warning symbol means danger. You are in a situation that could cause bodily injury. Before you work on any equipment, be aware of the hazards involved with electrical circuitry and be familiar with standard practices for preventing accidents. Use the statement number provided at the end of each warning to locate its translation in the translated safety warnings that accompanied this device.

SAVE THESE INSTRUCTIONS

# Related Documentation

The Application Centric Infrastructure documentation set includes the following documents that are available on Cisco.com at the following URL: http://www.cisco.com/c/en/us/support/cloud-systems-management/application-policy-infrastructure-controller-apic/tsd-products-support-series-home.html.

### Web-Based Documentation

- *Cisco APIC Management Information Model Reference*
- *Cisco APIC Online Help Reference*
- *Cisco APIC Python SDK Reference*
- *Cisco ACI Compatibility Tool*
- *Cisco ACI MIB Support List*

### Downloadable Documentation

- *Knowledge Base Articles* (KB Articles) are available at the following URL: http://www.cisco.com/c/en/us/support/cloud-systems-management/application-policy-infrastructure-controller-apic/tsd-products-support-series-home.html
- *Cisco Application Centric Infrastructure Controller Release Notes*
- *Cisco Application Centric Infrastructure Fundamentals Guide*
- *Cisco APIC Getting Started Guide*
- *Cisco ACI Basic Configuration Guide*
- *Cisco ACI Virtualization Guide*
- *Cisco APIC REST API User Guide*
- *Cisco APIC Object Model Command Line Interface User Guide*
- *Cisco APIC NX-OS Style Command-Line Interface Configuration Guide*
- *Cisco APIC Faults, Events, and System Messages Management Guide*
- *Cisco ACI System Messages Reference Guide*
- *Cisco APIC Layer 4 to Layer 7 Services Deployment Guide*
- *Cisco APIC Layer 4 to Layer 7 Device Package Development Guide*

- *Cisco APIC Layer 4 to Layer 7 Device Package Test Guide*
- *Cisco ACI Firmware Management Guide*
- *Cisco ACI Troubleshooting Guide*
- *Cisco APIC NX-OS Style CLI Command Reference*
- *Cisco ACI Switch Command Reference, NX-OS Release 11.0*
- *Verified Scalability Guide for Cisco ACI*
- *Cisco ACI MIB Quick Reference*
- *Cisco Nexus CLI to Cisco APIC Mapping Guide*
- *Application Centric Infrastructure Fabric Hardware Installation Guide*
- *Cisco NX-OS Release Notes for Cisco Nexus 9000 Series ACI-Mode Switches*
- *Nexus 9000 Series ACI Mode Licensing Guide*
- *Cisco Nexus 9332PQ ACI-Mode Switch Hardware Installation Guide*
- *Cisco Nexus 9336PQ ACI-Mode Switch Hardware Installation Guide*
- *Cisco Nexus 9372PX and 9372PX-E ACI-Mode Switch Hardware Installation Guide*
- *Cisco Nexus 9372TX ACI-Mode Switch Hardware Installation Guide*
- *Cisco Nexus 9396PX ACI-Mode Switch Hardware Installation Guide*
- *Cisco Nexus 9396TX ACI-Mode Switch Hardware Installation Guide*
- *Cisco Nexus 93128TX ACI-Mode Switch Hardware Installation Guide*
- *Cisco Nexus 9504 NX-OS Mode Switch Hardware Installation Guide*
- *Cisco Nexus 9508 ACI-Mode Switch Hardware Installation Guide*
- *Cisco Nexus 9516 ACI-Mode Switch Hardware Installation Guide*

### Cisco Application Centric Infrastructure (ACI) Simulator Documentation

The following Cisco ACI Simulator documentation is available at http://www.cisco.com/c/en/us/support/cloud-systems-management/application-centric-infrastructure-simulator/tsd-products-support-series-home.html.

- *Cisco ACI Simulator Release Notes*
- *Cisco ACI Simulator Installation Guide*
- *Cisco ACI Simulator Getting Started Guide*

### Cisco Nexus 9000 Series Switches Documentation

The Cisco Nexus 9000 Series Switches documentation is available at http://www.cisco.com/c/en/us/support/switches/nexus-9000-series-switches/tsd-products-support-series-home.html.

### Cisco Application Virtual Switch Documentation

The Cisco Application Virtual Switch (AVS) documentation is available at http://www.cisco.com/c/en/us/support/switches/application-virtual-switch/tsd-products-support-series-home.html.

### Cisco Application Centric Infrastructure (ACI) Integration with OpenStack Documentation

Cisco ACI integration with OpenStack documentation is available at http://www.cisco.com/c/en/us/support/cloud-systems-management/application-policy-infrastructure-controller-apic/tsd-products-support-series-home.html.

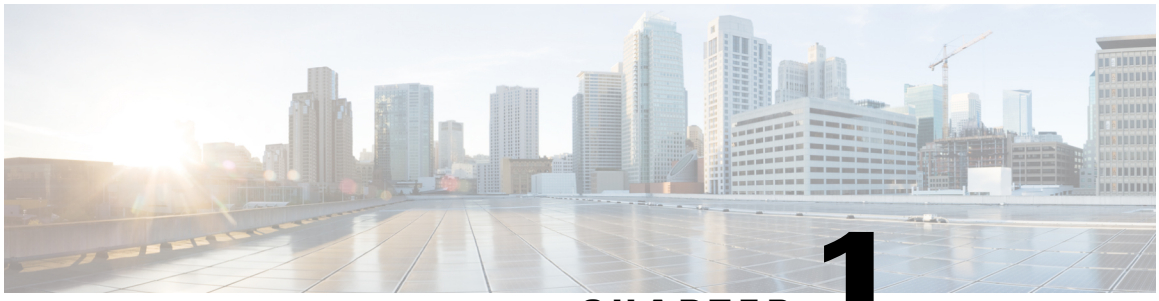
# Documentation Feedback

To provide technical feedback on this document, or to report an error or omission, please send your comments to apic-docfeedback@cisco.com. We appreciate your feedback.

# Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, using the Cisco Bug Search Tool (BST), submitting a service request, and gathering additional information, see *What's New in Cisco Product Documentation* at: http://www.cisco.com/c/en/us/td/docs/general/whatsnew/whatsnew.html

Subscribe to *What's New in Cisco Product Documentation*, which lists all new and revised Cisco technical documentation as an RSS feed and delivers content directly to your desktop using a reader application. The RSS feeds are a free service.



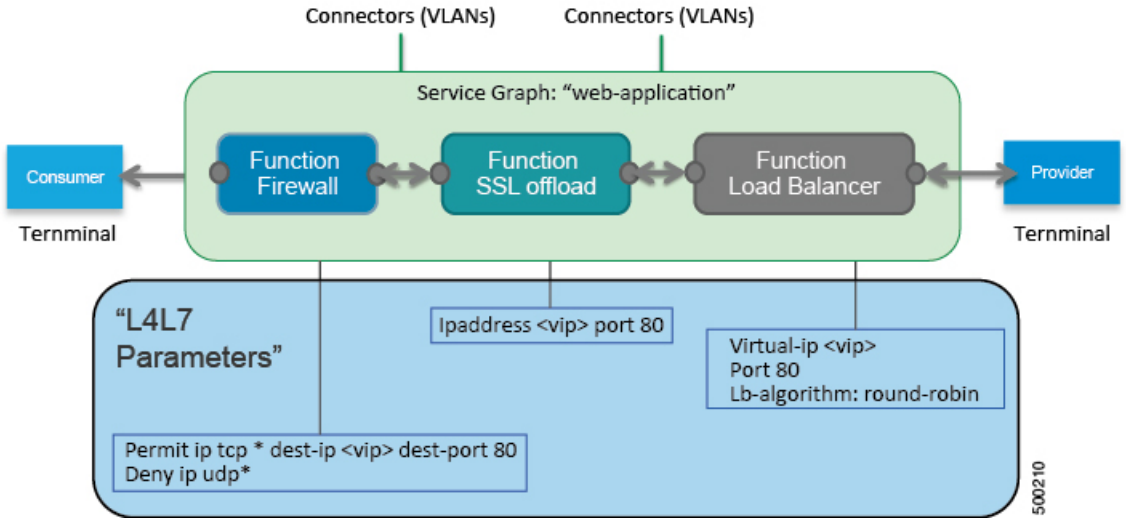
CHAPTER 1

# Overview

# About Service Graphs

A service graph is an order set of Layer 4 to Layer 7 devices between two endpoint groups.

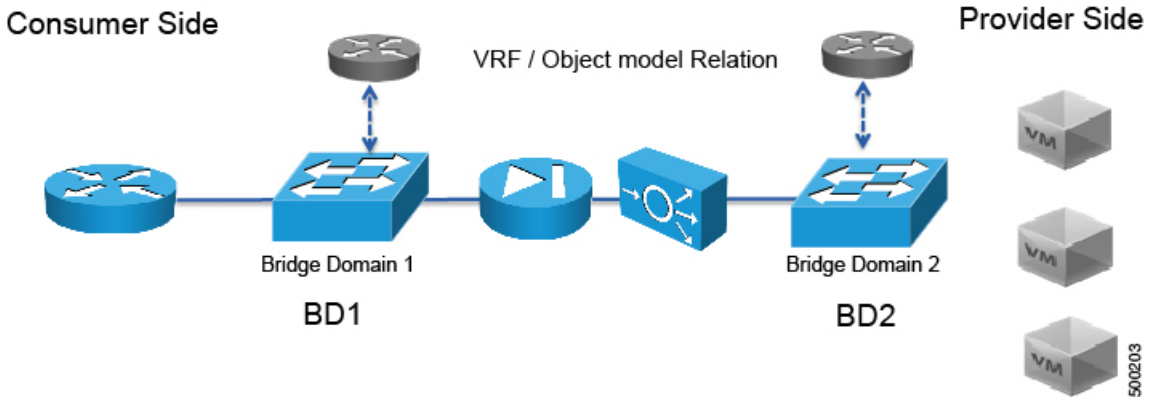
***Figure 1: Example Service Graph Deployment***




By using a service graph, you can install a service, such as the ASA firewall, once and deploy it multiple times in different logical topologies. Each time the graph is deployed, Cisco Application Centric Infrastructure (ACI) takes care of changing the configuration on the firewall to enable the forwarding in the new logical topology.

Deploying a service graph requires bridge domains and VRFs, as shown in the following figure:

***Figure 2: Bridge Domains and VRFs of a Service Graph***



# Advantages and Disadvantages of Using a Service Graph

Using a service graph provides several advantages and some disadvantages over not using one.

The advantages are as follows:

- Is a configuration template that can be reused multiple times
- Provides a more logical view and an application-related view of services
- Can provision a device that is shared across multiple departments
- Automatically manages VLAN assignments
- Automatically plugs vNICs
- Collects health scores from the device or service
- Collects statistics from the device
- Updates ACLs and pools automatically with endpoint discovery
- Can use unmanaged mode to avoid using a device package

The disadvantages are as follows:

- The topology is restricted; for example, the graph is always associated with a contract, which means it is always a producer-consumer relationship
- A multi-legged firewall deployment is more complex
- The operational model is orientated toward automation

# When to Use a Service Graph

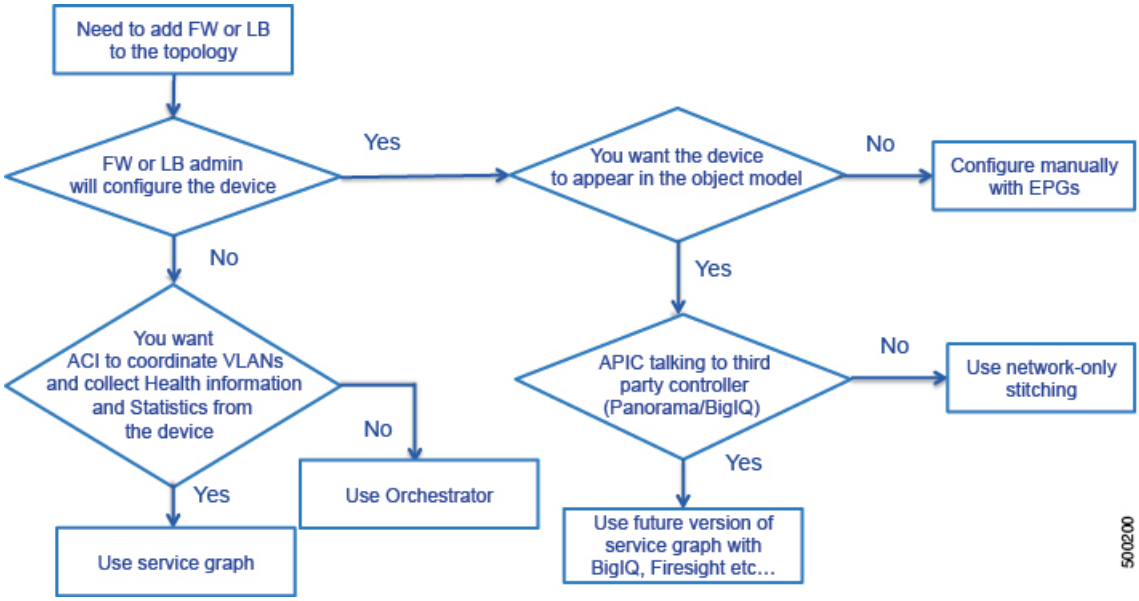
A service graph is most often used for the following things:

- Automation
- Integration of Cisco Application Centric Infrastructure (ACI) and services for advanced features

You do not need to use a service graph all of the time. For example, you might want to use unmanaged mode or you might only want to create endpoint groups and plug a firewall and load balancer into the endpoint groups. In such cases, you do not need a service graph.

The following flowchart can help you determine if you should use a service graph:

***Figure 3: When to Use a Service Graph***




# Methods for Configuring a Service Graph

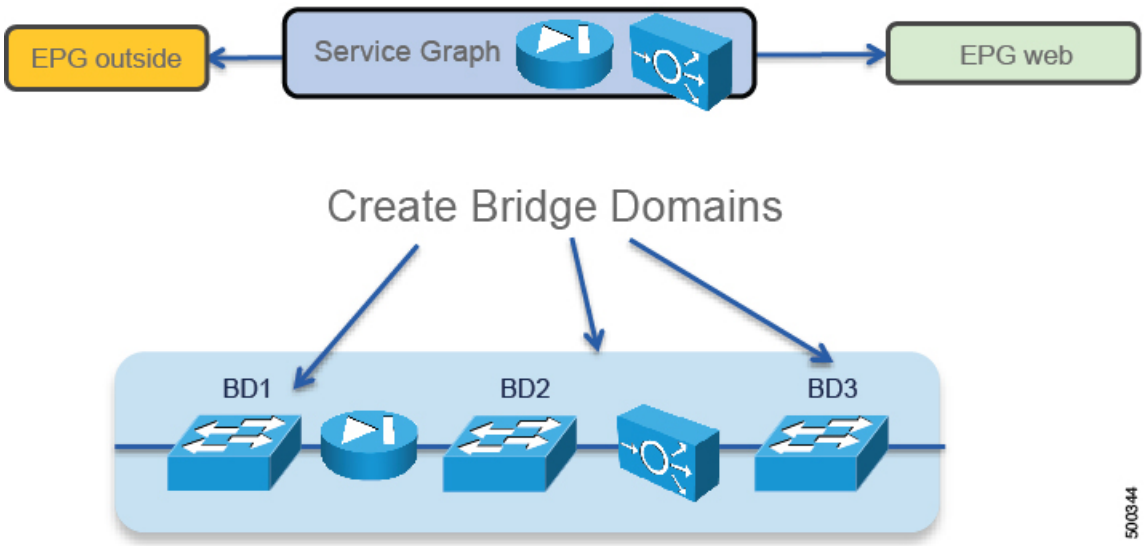
You can configure a service graph by using the following methods:

- GUI—If you are learning how to configure a service graph or if you are using predefined function profiles, you can use the GUI to validate the configuration and then save the service graph in XML format.
- REST API—Use the REST API in a production environment by integrating REST calls into Python scripts to automate the provisioning of a service graph.

# About Multi-Node Service Graphs

You can configure a multi-node service graph, which is a service graph that has more than one Layer 4 to Layer 7 service. The following figure illustrates the bridge domain configuration of a multi-node service graph:

***Figure 4: Bridge Domain Configuration of a Multi-Node Service Graph***




The bridge domains act as the links between the Layer 4 to Layer 7 devices.

**Note** The GUI enables you to configure multi-node service graphs consisting of 2 nodes, while the REST API enables you to configure up to 3 nodes in a single service graph.

# About the Service Graph Operational Model

You use a different operational model when you use a service graph compared to not using a service graph. Without a service graph, you use the following operational model:

- The network administrator configures the ports and VLANs to connect the firewall or the load balancer.
- On day 0, the firewall administrator configures the ports and VLANs.
- On day 1, the firewall administrator configures the ACLs and other components.
- The three configurations are spread over multiple days.

With a service graph, you use the following operational model:

- The Cisco Application Centric Infrastructure (ACI) administrator configures the ports and VLANs to connect the firewall or the load balancer.

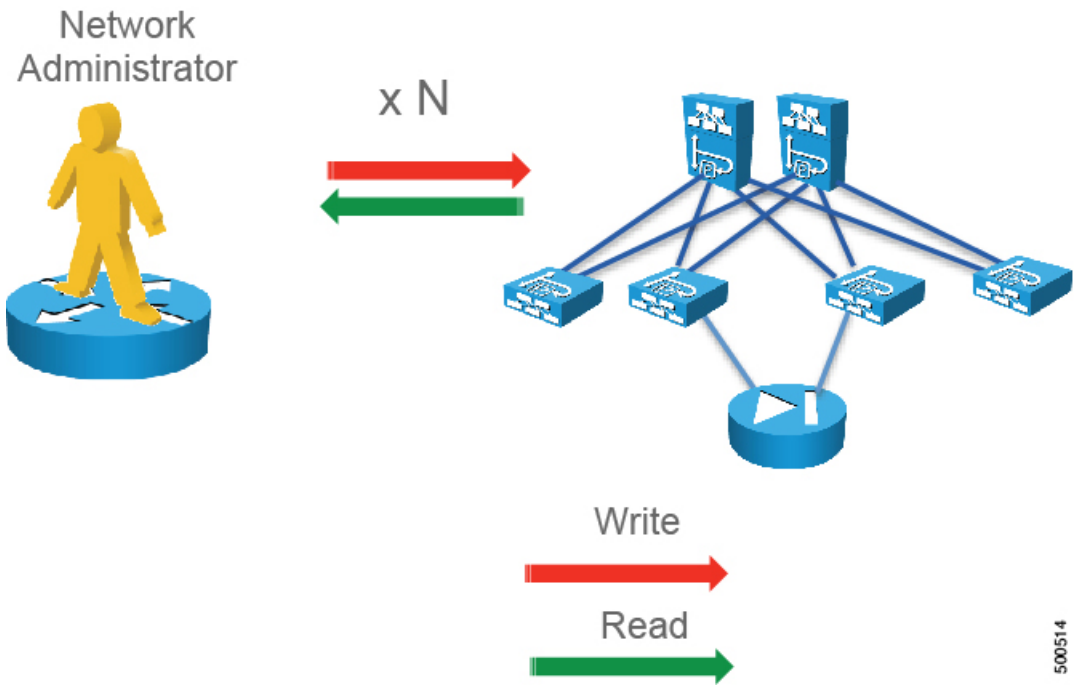
- The firewall administrator configures the ports, VLANs, ACLs, and other components.
- All configurations are performed in a single step.

The following figure illustrates how the operational model changes when you use a service graph:

***Figure 5: Service Graph Operational Model***


The following figure illustrates the operational model of network administration without Cisco Application Centric Infrastructure (ACI):

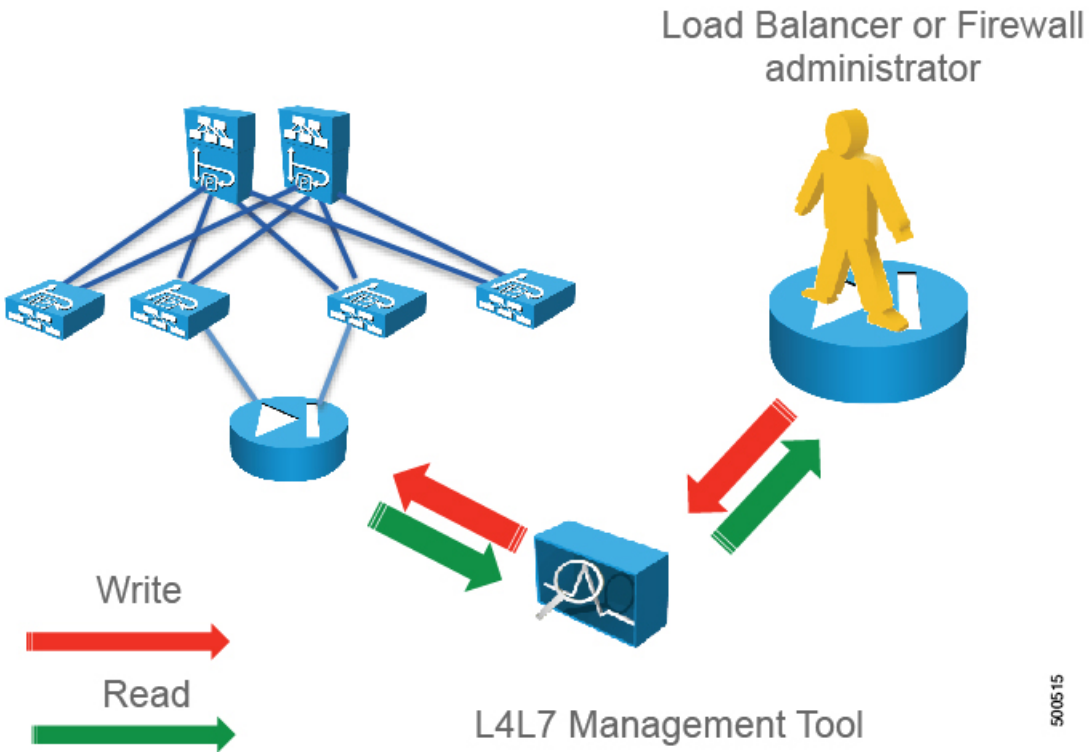
***Figure 6: Network Administration Without ACI***




The network administrator uses a network management tool to configure each individual network.

The following figure illustrates the operational model of Layer 4 to Layer 7 services administration without ACI:

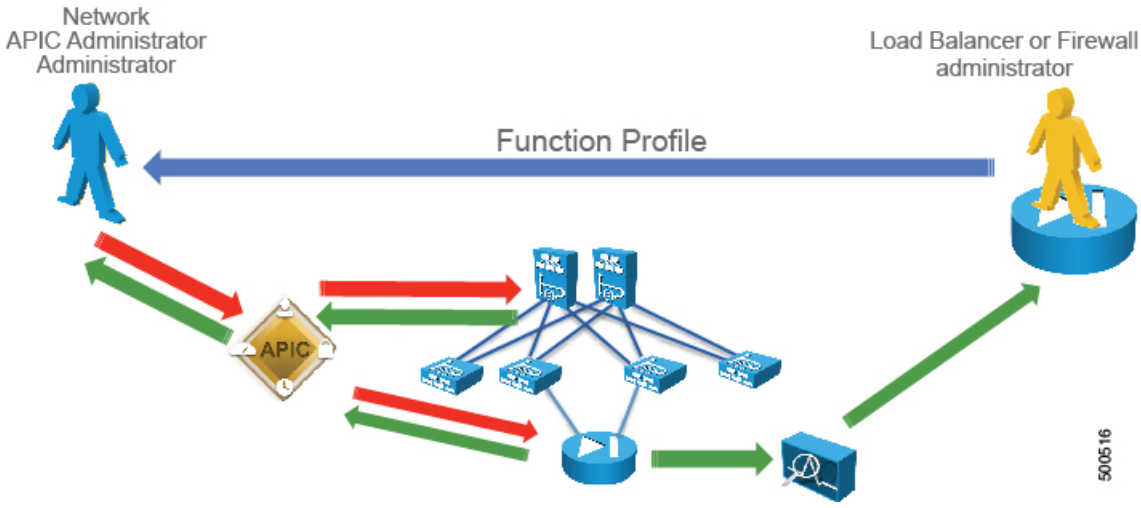
***Figure 7: Layer 4 to Layer 7 Services Administration Without ACI***




The load balancer or firewall administrator uses a Layer 4 to Layer 7 management tool to configure a load balancer or firewall for each individual network.

The following figure illustrates the operational model of Layer 4 to Layer 7 services administration with ACI:

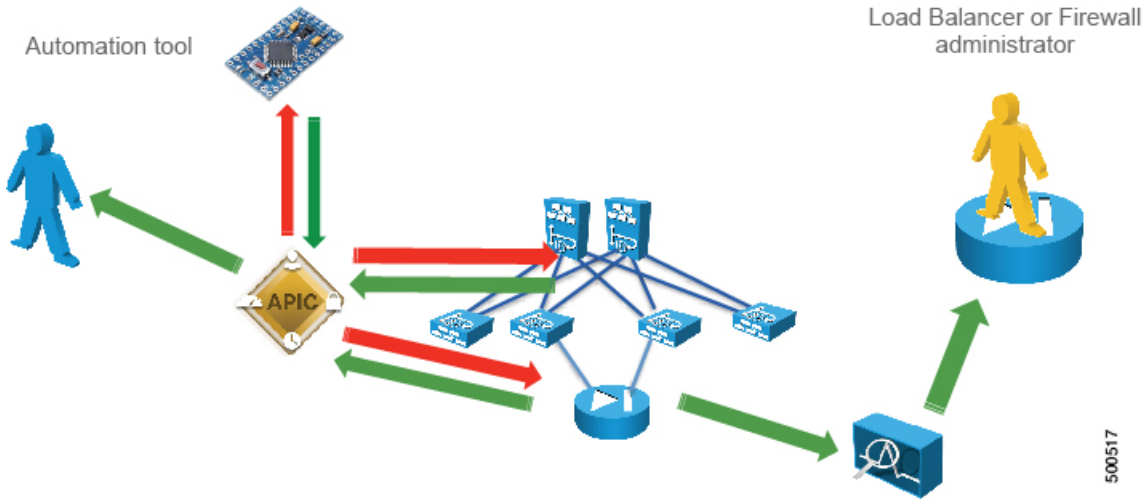
***Figure 8: Layer 4 to Layer 7 Services Administration with ACI***




The load balancer or firewall administrator creates a function profile that the Application Policy Infrastructure Controller (APIC) administrator uses in the APIC to configure a network, firewall, or load balancer. A function profile provides the default values for a service graph template.

The following figure illustrates the operational model of Layer 4 to Layer 7 services administration with ACI using automation:

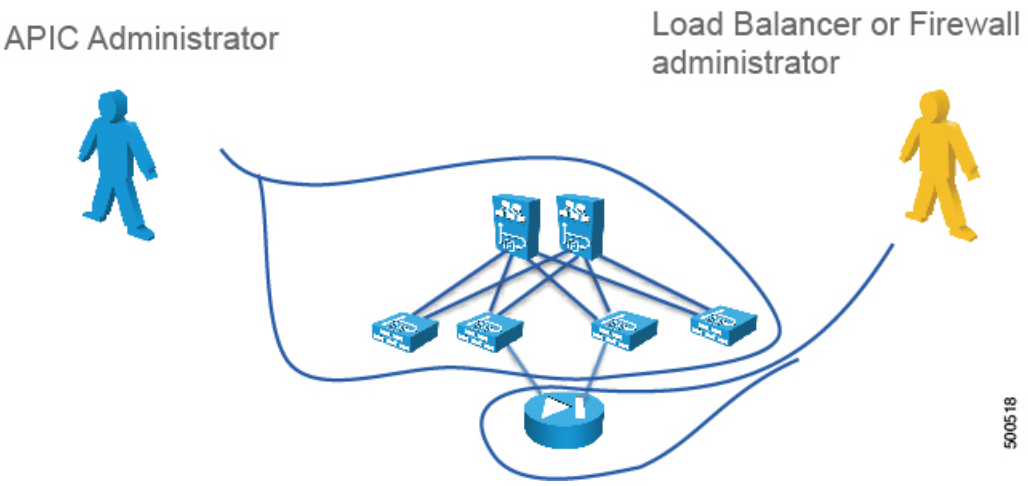
***Figure 9: Layer 4 to Layer 7 Services Administration with ACI Using Automation***




The APIC administrator uses an automation tool that uses a function profile to configure all networks, firewalls, and load balancers.

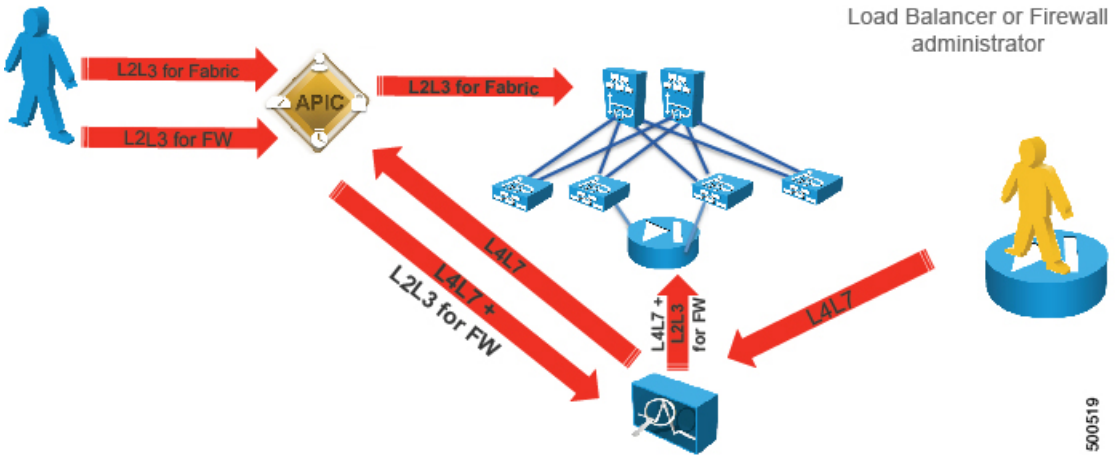
In all of these models, the APIC administrator wants control over the network, while the load balancer or firewall administrator wants control over the load balancer or firewall.

***Figure 10: What Administrators Want***




The solution to this issue is to deploy Layer 4 to Layer 7 services with a device manager, as shown in the following figure:

***Figure 11: Layer 4 to Layer 7 Services Administration with a Device Manager***




# About Goto Devices and GoThrough Devices

You can configure a logical device as one of the following function types:

- GoTo—The logical device is in routed mode.
- GoThrough—The logical device is in transparent mode, which is also known as bridged mode. A packet goes through without being addressed to the device, and endpoints are not aware of that device.

# About Contracts

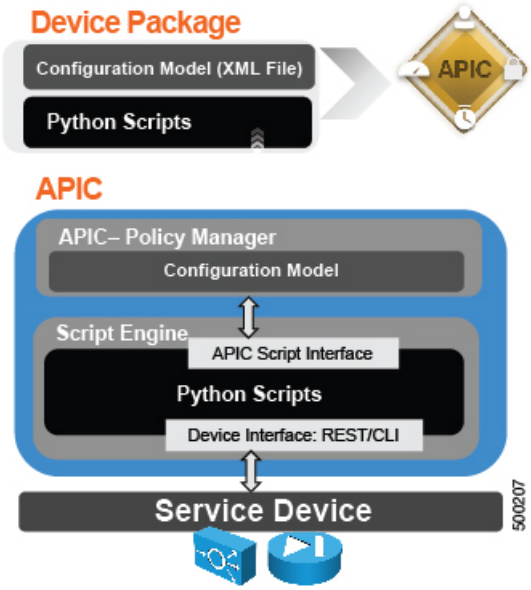
Contracts define inbound and outbound permit, deny, and QoS rules between endpoint groups. Contracts allow both simple and complex definition of the way that an endpoint group communicates with other endpoint groups. Contracts connect endpoint groups using a provider-consumer relationship. One endpoint group provides a contract and other endpoint groups consume that contract, and each endpoint group is associated with a bridge domain. The service graph is always associated with a contract, thus connecting a client-side (outside) or consumer endpoint group to a server-side (inside) provider endpoint group.

# About Device Packages

The Application Policy Infrastructure Controller (APIC) requires a device package to configure and monitor service devices. You add service functions to the APIC through the device package.

A device package contains a device configuration model and device scripts. A device configuration model is an XML file that defines a service function and configuration. A device script is a Python script that translates APIC API callouts to device-specific callouts. A device script can interface with the device by using REST, SSH, or any similar mechanism.

***Figure 12: Device Package and the APIC***

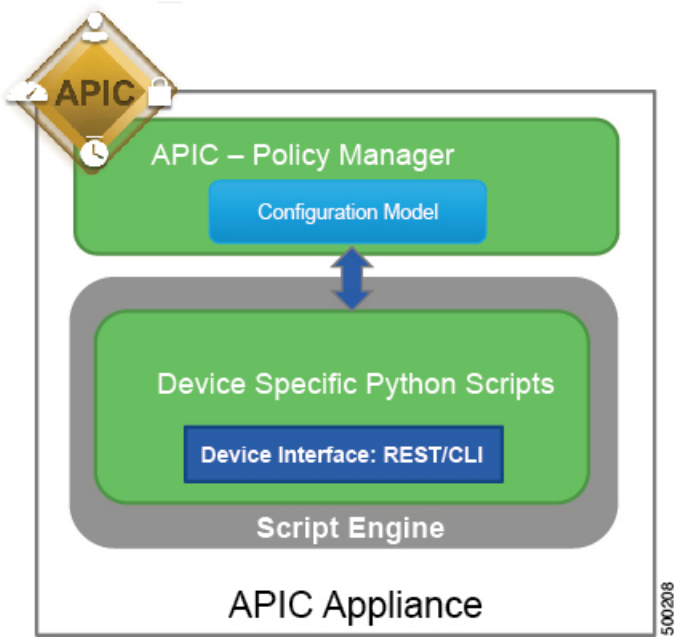



The functions in a device script are classified into the following categories:

- Device/Infrastructure—For device level configuration and monitoring
- Service Events—For configuring functions, such as a server load balancer or Secure Sockets Layer, on the device
- Endpoint/Network Events—For handling endpoint and network attach/detach events

The APIC uses the device configuration model that is provided in the device package to pass the appropriate configuration to the device scripts. The device script handlers interface with the device using its REST or CLI interface.

***Figure 13: How the Device Scripts Interface with a Service Device***




For more information about device packages and how to develop a device package, see *Cisco APIC Layer 4 to Layer 7 Device Package Development Guide*

# About Device Package Versions

Each device package has three types of versions:

- Major Version—Multiple major versions can co-exists in an Application Policy Infrastructure Controller (APIC). For example, ACME 9000 and ACME Chassis-10000 can have different packages with different major versions.

  The following XML string shows a major version:

  ```
  <vnsMDev vendor="Acme" model="ADC" version="10.5">
  ```

- Minor Version—Represents a different version of the packages for the same major version. Only one minor version can be active in an APIC at a given time. The minor version is used to do versioning of software releases of a device package for a specific major version.

- ctrlrVersion—When a package is developed, it is developed against a specific APIC version. The APIC validates this in the policymgr against the APIC's running version. The package upload fails if there is a mismatch.

  The following XML string shows a minor version and ctrlrVersion:

  ```
  <vnsDevScript name="Acme" packageName="AcmeDeviceScript.py"
    minorversion="10.51" ctrlrVersion="1.0"/>
  ```

# About Device Package Upgrades

You can upgrade a device package by uploading a new one to the Application Policy Infrastructure Controller (APIC). The device package version is a concatenation of the controller version and device package minor version. The first part of the device package version (1.0 in the ASA example) should be greater than or equal to the APIC version.

If the major version (the naming property of class `vnsMDev`) changes, uploading the device package will create a new device package. For example, if the original ASA package distinguished name was "uni/infra/mDev-CISCO-ASA-1.0" and the new package version changed to "2.0", then the new distinguished name will be "uni/infra/mDev-CISCO-ASA-2.0". The system will have two packages; the old service graphs and device clusters will continue to point to the old package and continue working. New service graphs and device clusters can use the old or new device package. Switching the old service graphs and device clusters to the new package will be disruptive.

Changing the minor version (a property called `minorversion` in the DevScript managed object) does not change the distinguished name of the package or `vnsMDev`. Uploading a new device package with a different `minorversion` overwrites the existing device package. All service graphs and device clusters that pointed to the old device package start pointing to the new device package automatically. The upgrade is non-disruptive and there should be no impact for existing service graphs or device clusters. A minor version change is the default recommendation for partners for any new package revisions.

When the APIC identifies that only the minor version has changed and that the device package version has not incremented, the APIC takes the following actions:

- Existing service graph instances using the existing device packages are terminated
- The script wrapper process hosting the device script is terminated and a new script wrapper process is initiated
- New graph instances are created
- Device audit and service audit is invoked on all of the new graph instances

The following table provides a recommendation for whether a device package change should be a major version or minor version change:

| Type of Change in Device Package | Recommended Upgrade Type |
|---|---|
| Script bug fixes | Minor version |
| Addition of any kind, such as new functions, folders, parameters, or profiles | Minor version |
| Modification or removal of any kind, such as functions, folder/parameters, or profiles | Major version |

APIC images are backward compatible with old device packages. If a device package is already uploaded and an APIC is upgraded, the old device package continues to work without any disruption. Newer device packages might not work on older versions; in such cases, the device package upload step fails with an appropriate error.

# About Virtual Appliances and Physical Appliances

The following table compares virtual appliances and physical appliances:

| Virtual Appliance | Physical Appliance |
|---|---|
| Cannot trunk on the vNIC | Supports trunking on the physical interfaces |
| If you need to use the service graph across multiple bridge domains:<br>• The appliance must have multiple vNICs<br>• You need more virtual appliances | The service graph can be re-used across multiple bridge domains |

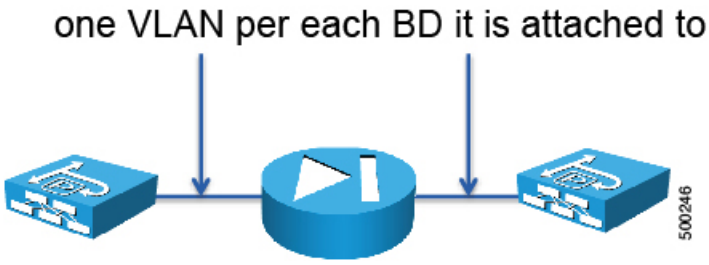
### Virtual Appliances

vNICs are automatically assigned to the port-groups. VLANs are automatically created on the ACI interfaces and on the Layer 4 to Layer 7 device. You cannot reuse the same graph on different bridge domains. There is no trunking on the vNICs, as shown in the following figure:


A service graph with virtual appliances works with virtual appliances running on a VMware vSphere Distributed Switch (VDS) or Cisco Application Virtual Switch (AVS) with VLANs.

### Physical Appliances

When you deploy a physical appliance, VLANs are automatically created on the Cisco Application Centric Infrastructure (ACI) interfaces and on the Layer 4 to Layer 7 device. One VLAN gets created for each bridge domain to which the physical appliance is attached, as shown in the following figure:



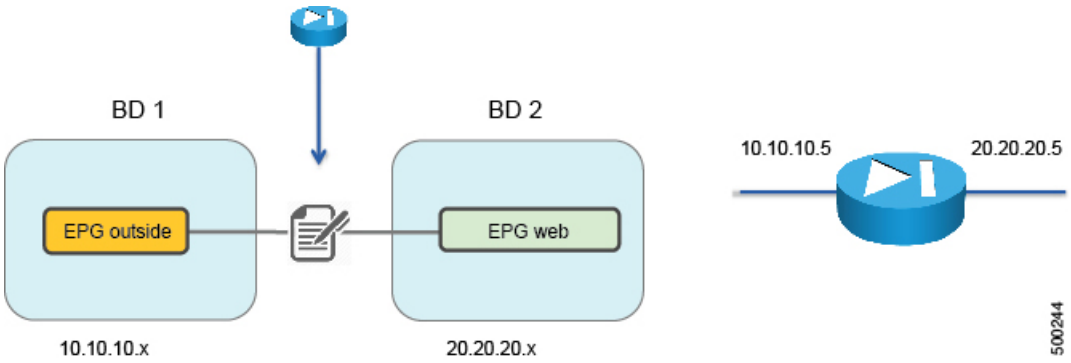
# Dataplane

## About Deployment Modes

There are three main deployment modes for a service graph:

- GoTo—The Layer 4 to Layer 7 device is a Layer 3 device that routes traffic; it is the default gateway for servers or the next hop
- GoThrough—The Layer 4 to Layer 7 device is a transparent Layer 2 device; the next-hop or the outside bridge domain provides the default gateway
- One-arm—The bridge domain of the servers is the default gateway

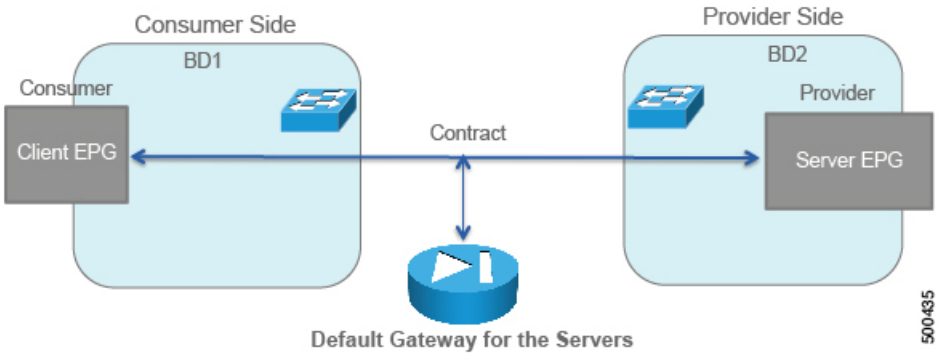
Except for one-arm mode, you must start with two bridge domains, as shown in the following figure:




### GoTo Mode

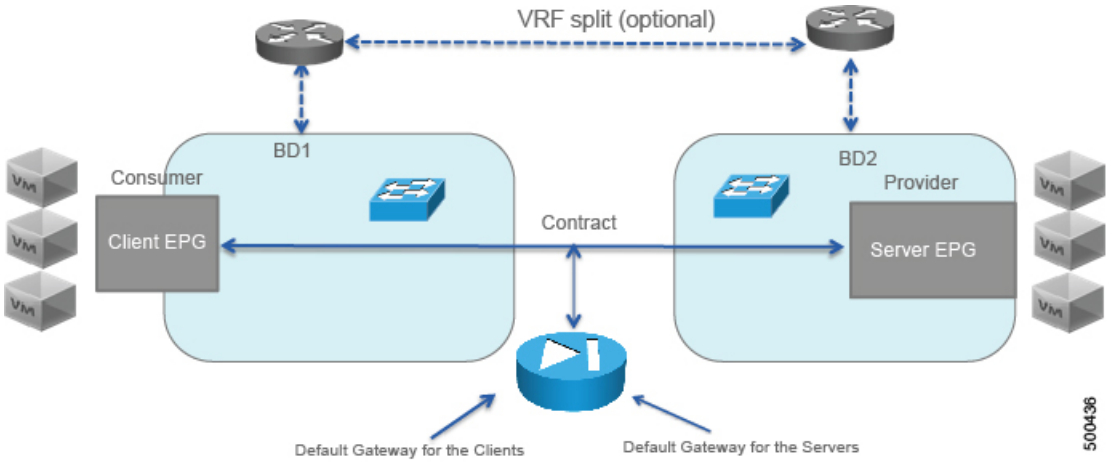
The following figure illustrates a generic GoTo mode deployment with its basic building blocks:

***Figure 14: GoTo Mode Deployment***



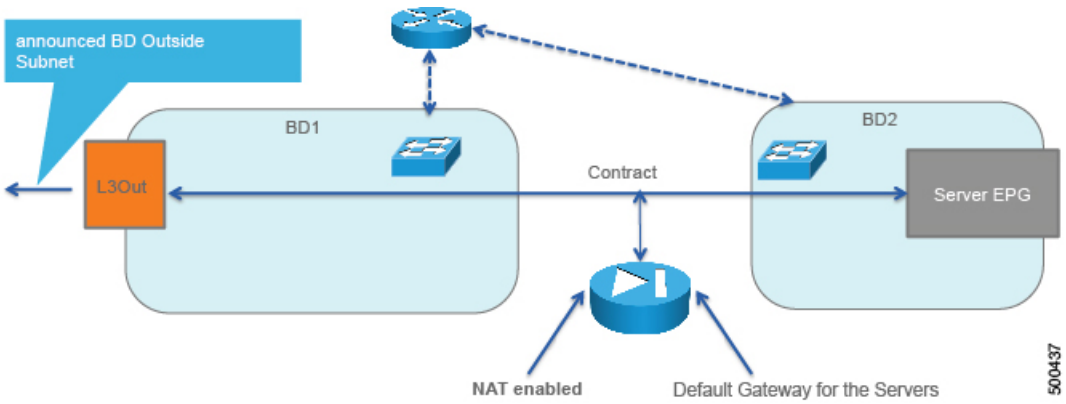
The following figure illustrates a GoTo mode deployment with client virtual machines, including the fact that you must provision VRFs to be associated with the bridge domains:

***Figure 15: GoTo Mode Deployment with Client Virtual Machines***




The following figure illustrates a GoTo mode deployment with a Layer 3 Outside (L3Out):

***Figure 16: GoTo Mode Deployment with a Layer 3 Outside***

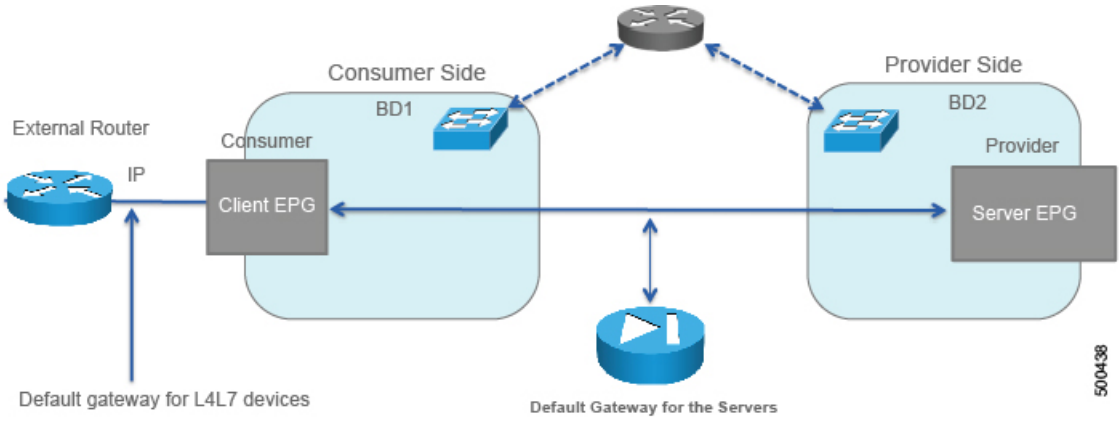



The L3Out consumes the contract through the Layer 3 external endpoint group, which is called `L3InstP` in the object model.

The bridge domain has a "Rs" with the L3Out, which indicates that in the bridge domain configuration, you must indicate with which L3Out the bridge domain is associated.

The following figure illustrates a GoTo mode deployment with an external router:

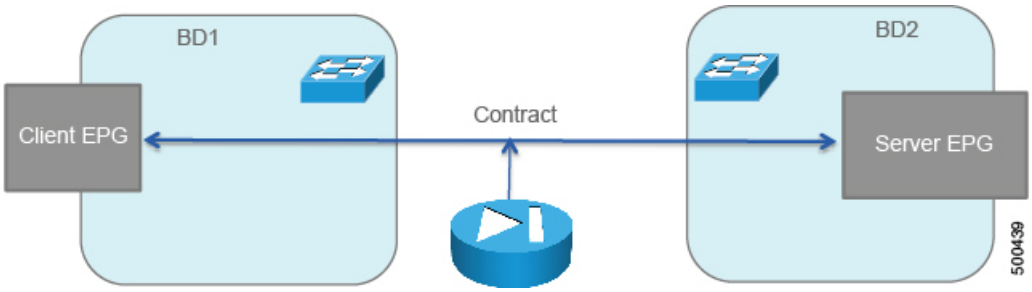
***Figure 17: GoTo Mode Deployment with an External Router***




## GoThrough

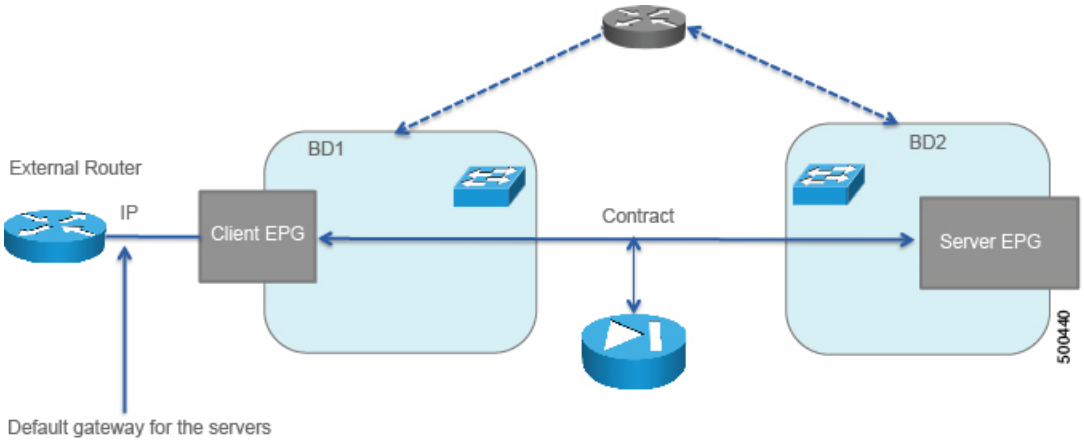
The following figure illustrates a GoThrough mode deployment with its basic building blocks:

***Figure 18: GoThrough Mode Deployment***



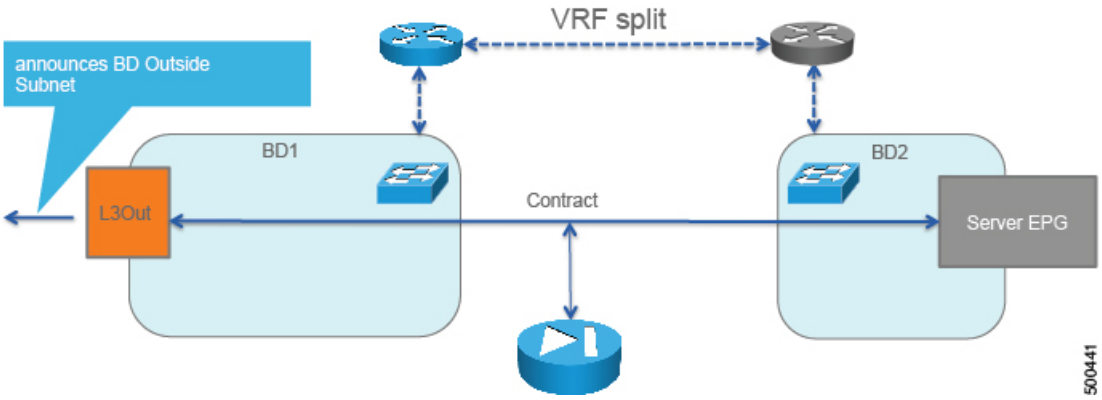
The following figure illustrates a GoThrough mode deployment with an external router, including the fact that you must provision VRFs to be associated with the bridge domains:

***Figure 19: GoThrough Mode Deployment with an External Router***




The following figure illustrates a GoThrough mode deployment with a Layer 3 Outside (L3Out):

***Figure 20: GoThrough Mode Deployment with a Layer 3 Outside***

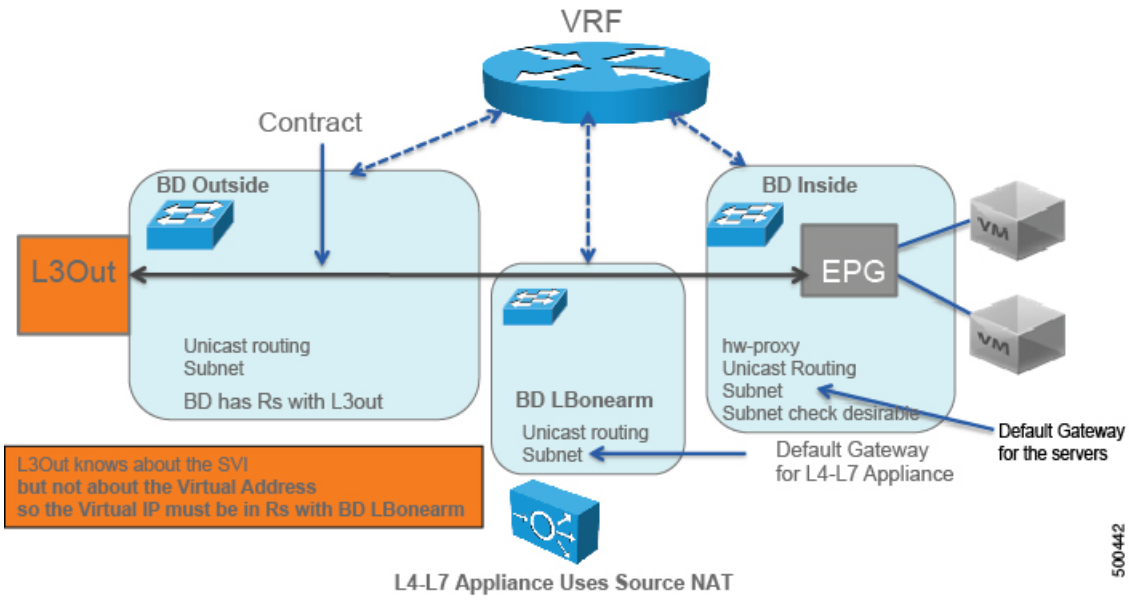



The L3Out consumes the contract through the Layer 3 external endpoint group, which is called `L3InstP` in the object model.

### One-Arm Mode

The following figure illustrates a one-arm mode deployment:

***Figure 21: One-Arm Mode Deployment***




The topology for a one-arm mode deployment is as follows:

- You need 3 bridge domains
  - One bridge domain for client side (external)
  - One bridge domain for the server side
  - One bridge domain for the load balancer, only

    The service graph template will create the association with the bridge domain
- The bridge domains are all enabled for unicast routing
- The subnet on the server side bridge domain is the default gateway for the servers
- The subnet on the load balancer bridge domain is the default gateway for the load balancer

# About Configuring Bridge Domains

With a service graph, you must configure a bridge domain for the client-side/consumer-side/outside, a bridge domain for the server-side/provider-side/inside, and bridge domains to stitch devices.

If you do not know which bridge domain settings to use, you can use the following:

- Unknown unicast flooding
- ARP flooding
- No IP routing

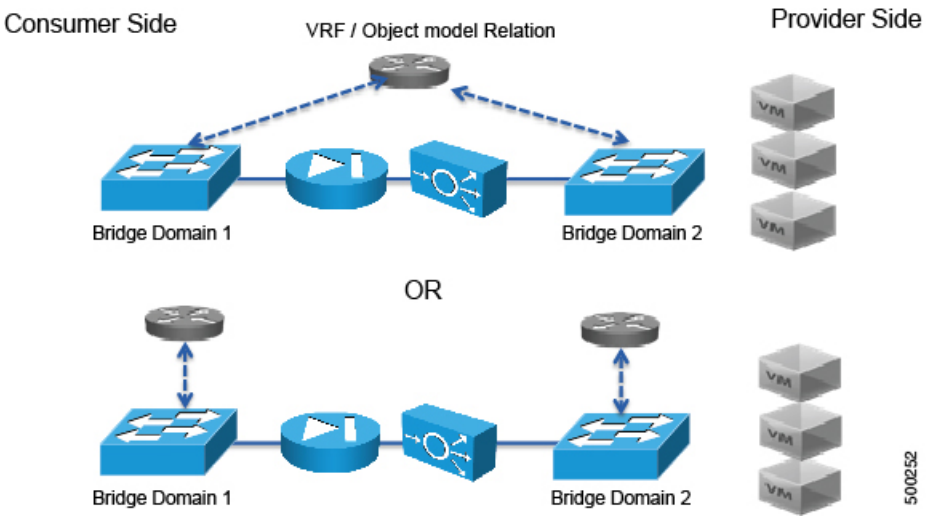
- No subnet

This is a valid configuration, but it might not be the best configuration for your setup. For more information about how to optimize the bridge domain configurations, see Deploying F5, on page 51 and Deploying ASA, on page 81.

# Determining the Number of VRFs to Use

In Cisco Application Centric Infrastructure (ACI), each bridge domain must always be associated to a VRF for the purpose of meeting the object model requirements. Each VRF has one or more bridge domains associated with it and when the bridge domains are configured for routing, the traffic of one bridge domain can be routed to another bridge domain of the same VRF. You must therefore determine how many VRFs to use when deploying the service graph.

***Figure 22: VRF and Object Model Relation***

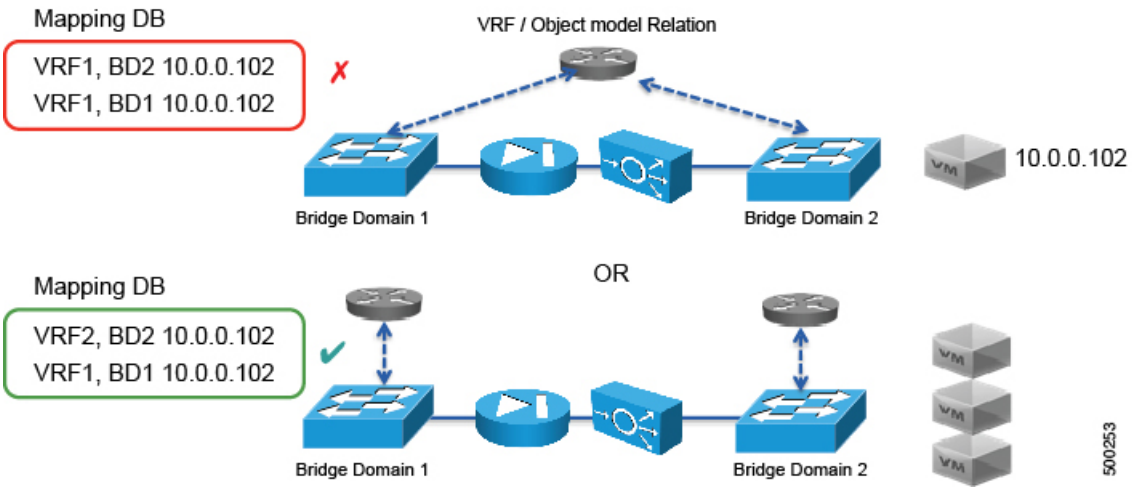



To decide how many VRFs that you need, you must understand how endpoint IP learning works:

- If routing is disabled under the bridge domain, then ACI only learns the MAC address.
- If routing is enabled under the bridge domain, then ACI learns the MAC address with Layer 2 traffic. ACI also learns the IP address when the host sends an ARP request to another host.
- Layer 2 traffic forwarding is still only based on the DMAC.
- The mapping database learns the IP address, which can be useful for troubleshooting and other functions.

The following figure illustrates how the mapping database gets programmed if 10.0.0.102 sends traffic:

***Figure 23: Programming the Mapping Database***




As Figure 23: Programming the Mapping Database, on page 20 shows, if you are deploying services and you have IP routing enabled on both bridge domains (1 and 2), you must create a VRF for each bridge domain that has IP routing enabled.

# About the Subnet Check

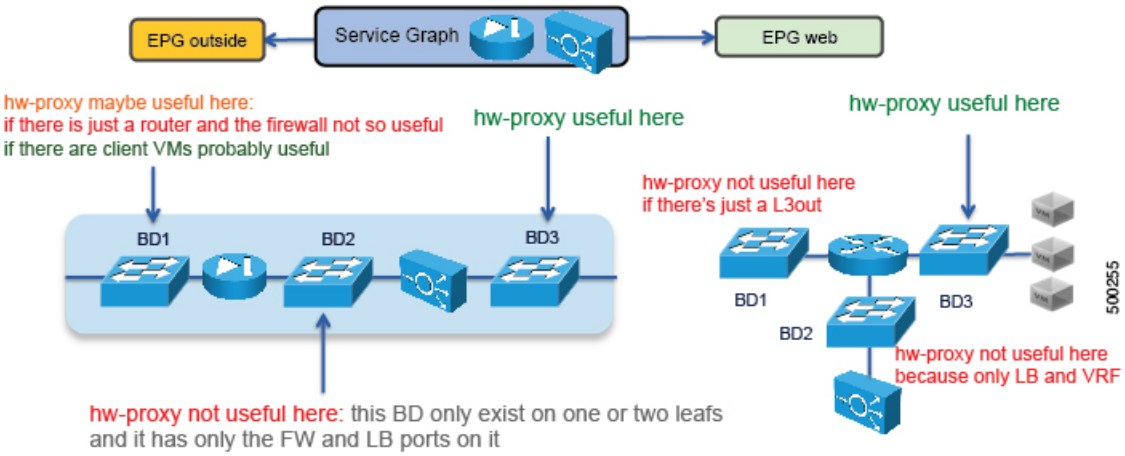
You can enforce the subnet check for IP address learning, which causes ACI to learn only IP addresses from configured subnets. This limits the IP address learning in the bridge domain to the IP addresses of the subnet that is specified. The other IP addresses are still forwarded, but are based on their MAC addresses and are not based on their IP addresses.

# About Hardware Proxy

The hardware proxy feature reduces flooding for Layer 2 unknown unicast packets. If the Layer 4 to Layer 7 appliance must be able to see flooded packets, then you cannot use hardware proxy. Otherwise, you can enable the hardware proxy on bridge domains that span multiple leafs to reduce the amount of flooded packets. Tuning the bridge domain to reduce packet flooding is beneficial when deploying the service graph in GoTo mode. When using GoThrough mode, Cisco Application Centric Infrastructure (ACI) automatically sets the bridge domains in unknown unicast flooding mode.

The following figure provides some examples of bridge domains for which hardware proxy can be beneficial:

***Figure 24: When to Use Hardware Proxy***




# About Multicontext Support

Multicontext support enables the same physical appliance to be exported to multiple tenants. You can create multiple partitions with a virtual appliance, but the vNICs cannot be shared because the virtual appliance is on multiple tenants, which means that there cannot be a trunk with VLANs on the same vNIC.

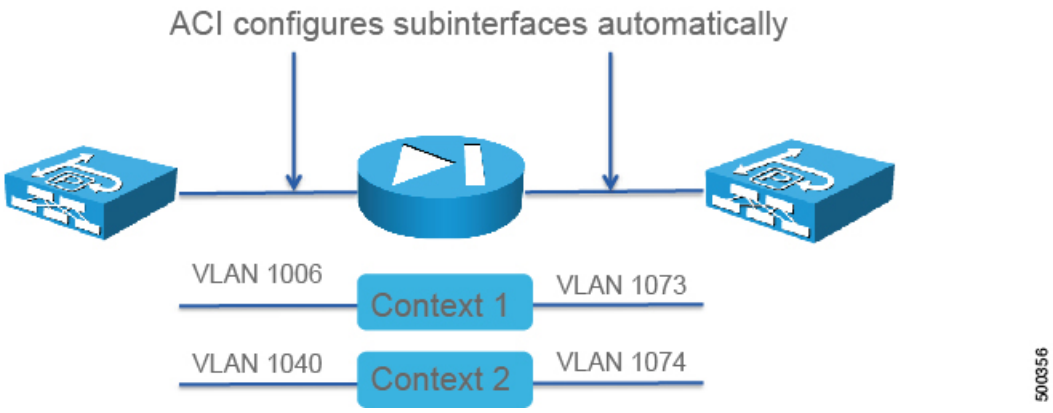
With ASA, you can partition a single physical ASA into multiple virtual firewalls, known as security/virtual contexts. Each context acts as an independent device with its own security policy, interfaces, and management IP address. The Application Policy Infrastructure Controller (APIC) does not create the ASA contexts; they must be predefined. Allocate-interface on the system context, firewall configuration on a virtual context, and the Cisco Application Centric Infrastructure (ACI) fabric policy are done by the APIC. The APIC needs to communicate with the system context and each virtual context.

With F5, partitions are automatically created and ACI tenants are automatically mapped to an F5 partition.

# About Multicontext Support and Dataplane Separation

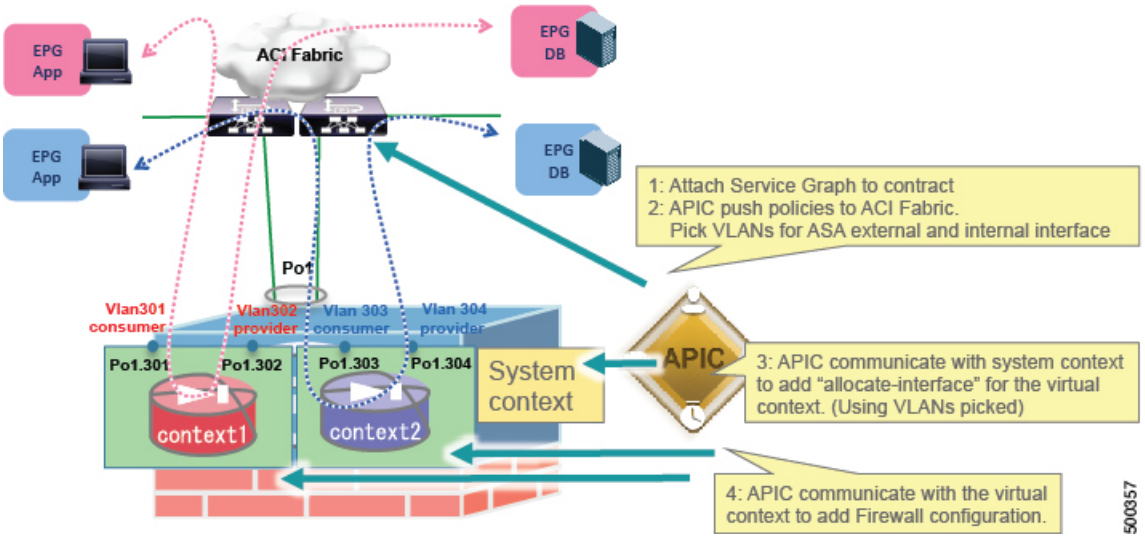
The Application Policy Infrastructure Controller (APIC) creates sub-interfaces based on a dynamically allocated VLAN from a pool, and in the system context it assigns port-channel sub-interfaces to appropriate user contexts. The following figure illustrates dataplane separation:

***Figure 25: Dataplane Separation***




The following figure illustrates how Cisco Application Centric Infrastructure (ACI) manages a multi-context ASA firewall:

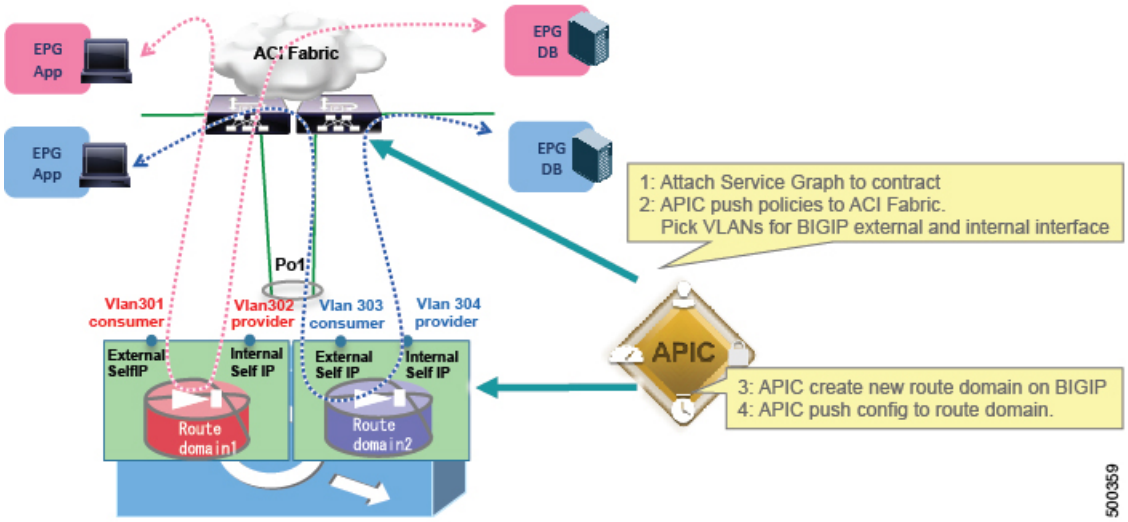
***Figure 26: How ACI Manages a Multi-Context ASA Firewall***




In the case of the ASA firewall, the APIC does not create the virtual context; they must be predefined. Allocate-interface, firewall configuration on a virtual context, and the ACI fabric policy are done by the APIC. The APIC needs to communicate with the system context and each virtual context.

The following figure illustrates how ACI manages multiple contexts with F5 BIGIP:

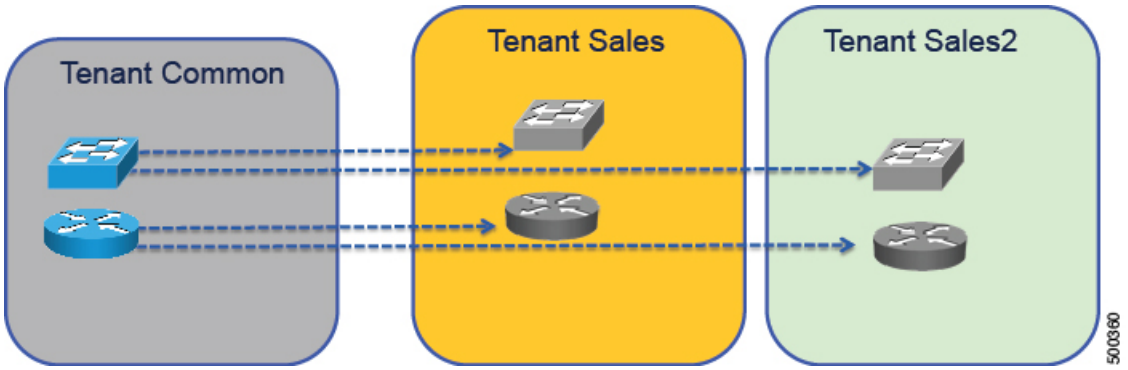
*Figure 27: How ACI Manages Multiple Contexts with F5 BIG-IP*




# About Sharing Service Devices

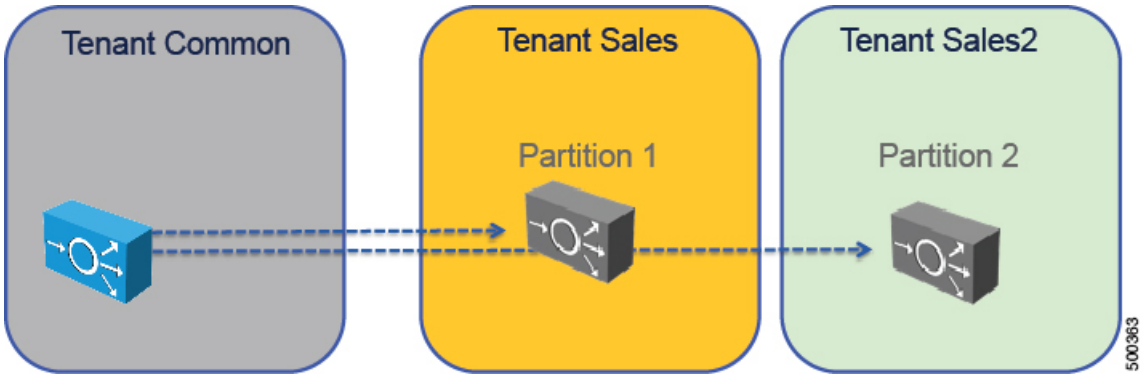
Cisco Application Centric Infrastructure (ACI) lets you configure objects in tenant `Common` that can be used by other tenants. Some of the objects include filters, bridge domains, VRFs, logical devices, and concrete devices. Tenants can attach endpoint groups to these objects. The following figure illustrates other tenants using objects that are configured in tenant `Common`:

*Figure 28: Sharing Tenant Common Objects*



With multicontext devices, you can share a device that is defined in tenant `Common` and use it from more than one tenant.

***Figure 29: Sharing Tenant Common Devices***

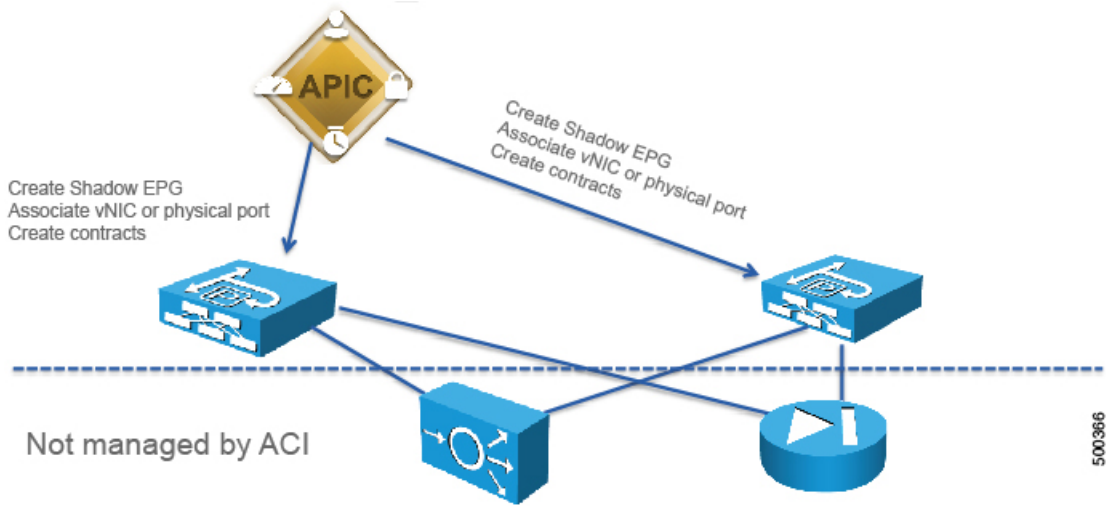



In addition to using tenant `Common` to share devices, you can also export contracts and Layer 4 to Layer 7 devices from any tenant for other tenants to use.

# About Unmanaged Mode

You can define a Layer 4 to Layer 7 service as unmanaged. With the unmanaged mode, Cisco Application Centric Infrastructure (ACI) only configures the fabric, not the Layer 4 to Layer 7 device.

***Figure 30: Unmanaged Mode***




For more information about the unmanaged mode, see *Cisco APIC Layer 4 to Layer 7 Services Deployment Guide* .

# Other Terminology

This section provides a high level description of other terminology that is not discussed in great detail in this document.

**Concrete Device**

Represents a service device, such as one load balancer or one firewall. It can be physical or virtual. Concrete devices are the devices member of a cluster, appear as child of the logical device, and own the physical or virtual interfaces. A concrete device defines concrete interfaces and maps an interface to a virtual adapter or path. A concrete device defines device-wide parameters, such as an HA or cluster configuration for ASA.

**Logical Device**

Represents a cluster of 2 devices that operate in active/standby mode. A logical device defines logical interfaces. The logical interface type is defined in the device model, and logical interfaces are used for the device selection policy. A logical device also defines cluster-wide parameters where applicable, such as NTP and DNS.

**Note** HA configuration is done on the concrete device on ASA.

**Function Node**

These nodes are functions of a device that are defined by the device package. A function node defines the configuration options that are available. Service graphs can contain one or more function nodes.

**Function Node Connector**

Each function node connector is allocated a VLAN by Cisco Application Centric Infrastructure (ACI); you do not need to manage that VLAN. A function node connector must be associated with a bridge domain. The user must have predefined the bridge domain and reference it in the configuration. Each side of function node connector is treated as an endpoint group and ACI automatically creates the endpoint group (shadow endpoint group) and puts a contract between the shadow endpoint group and the endpoint group created by the user.

**Logical Device Context**

Also known as a device selection policy, a logical device context selects the appropriate logical device and interfaces based on the following selectors:

- Service graph template name
- Contract name
- Node name

**Service Graph Connection**

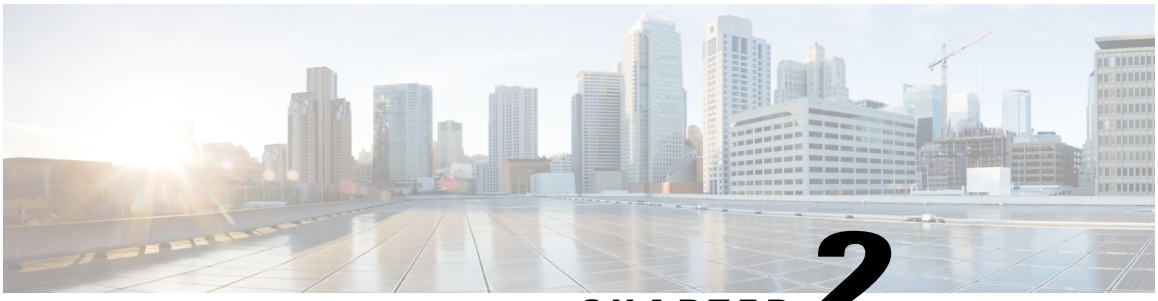
Service graph connections are used when multiple nodes are link. They act as a cable in-between 2 nodes (`AbsFConn`). The cable can be Layer 2 or Layer 3, with or without unicast routing.

### Service Graph Template

A generic representation of the expected traffic flow that defines connection points (connections and terminals) and the sequence of nodes and functions. A service graph template must be applied for it to be rendered.

### Terminal

Terminals define the consumer (`AbsTermNodeCon`) and provider (AbsTermNodeProv) links to the contract. Terminals need a connection to the node.



CHAPTER 2

# Supported Devices



## ADC Device Package Support

The following table provides the device package support for the application delivery controller (ADC) at the time of this writing:

| | Virtual/ Physical | Mode | Function Profile | HA | Multi-context on physical appliance | Dynamic Routing | Dynamic EPG | IPv6 | Feature | Operational Model |
|---|---|---|---|---|---|---|---|---|---|---|
| Citrix NetScaler | Both | GoTo (one-arm and two-arm) | Yes | No (manual OOB) | Yes; create virtual instance on SDX manually | Yes | Yes; member of pool for VIP | Yes | ADC | Everything through APIC |
| F5 BIG-IP LTM | Both | GoTo (one-arm and two-arm) | Yes | Yes | Yes; create route-domain on physical LTM automatically or create vCMP manually (no HA) | No | Yes; member of pool for VIP | No | ADC | Everything through APIC or BIG-IQ |
| A10 Thunder | Both | GoTo (one-arm and two-arm) | No | No (manual OOB) | No | No | No | No | ADC | Everything through APIC |

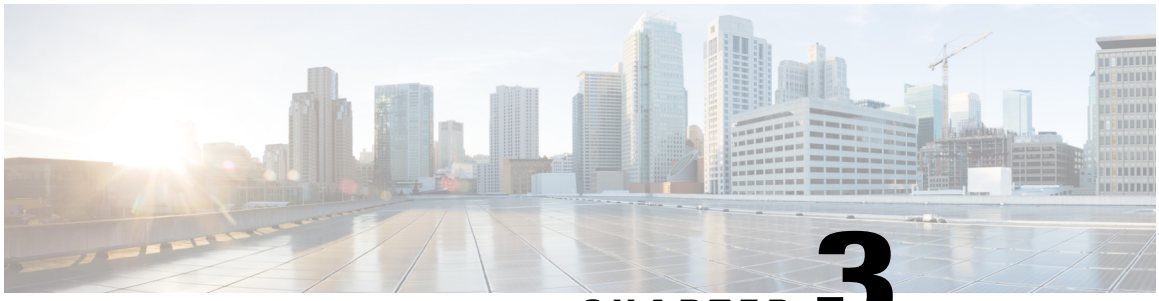
| | Virtual/ Physical | Mode | Function Profile | HA | Multi-context on physical appliance | Dynamic Routing | Dynamic EPG | IPv6 | Feature | Operational Model |
|---|---|---|---|---|---|---|---|---|---|---|
| Radware Alteon | Physical | GoTo | No | No | No | No | No | No | ADC | Everything through APIC |

# Firewall Device Package Support

The following table provides the device package support for firewalls at the time of this writing:

| | Virtual/ Physical | Mode | Function Profile | HA | Multi-context on physical appliance | Dynamic Routing | Dynamic EPG | IPv6 | Feature | Operational Model |
|---|---|---|---|---|---|---|---|---|---|---|
| Cisco ASA 5585 and ASAv30 | Both | GoTo, GoThrough | Yes | Yes | Yes; create context on ASA5500X manually<br>Allocate-interface to each context is done by APIC | Yes | Yes; object-group for ACE | Yes | FW, ACL, NAT | Everything through APIC |
| Cisco FirePOWER | Both | GoThrough | Yes | No | No | No | No | No | IPS | Everything through APIC or BIG-IQ |
| F5 | Both | GoTo, One-arm | Yes | Yes | Yes, through partitions<br>vCMP without HA | No | No | Yes | | Everything through APIC or Big-IQ |
| Citrix Netscaler | Both | GoTo, One-arm | Yes | No | Yes; create instances manually | Yes | Yes | Yes | | Everything through APIC |
| A10 | Both | GoTo, One-arm | Yes | Yes | No | No | No | No | | Everything through APIC |
| Radware | | GoTo | Yes | No | No | No | No | No | | Everything through APIC |

| | Virtual/ Physical | Mode | Function Profile | HA | Multi-context on physical appliance | Dynamic Routing | Dynamic EPG | IPv6 | Feature | Operational Model |
|---|---|---|---|---|---|---|---|---|---|---|
| Avi Networks | Virtual | GoTo | Yes | Yes | No | No | No | No | | Avi controller |



CHAPTER 3

# Deploying a Service Graph



## Overview of Deploying a Service Graph

The following list provides an overview of the tasks that you must perform to deploy a service graph:

1. Create physical and virtual domains.
2. Configure the basic management access on the Layer 4 to Layer 7 device
3. Import the device package.

   See Importing a Device Package Using the GUI, on page 35.
4. Create the bridge domains and VRFs.

   See Creating Bridge Domains and VRFs Using the GUI, on page 36.
5. Create endpoint groups and contracts.

   See Creating Endpoint Groups and Contracts Using the GUI, on page 37.
6. Configure logical devices and concrete devices.

See Creating a Logical or Concrete Device Using the GUI, on page 39.

7 Create or import a function profile.

See Creating a Function Profile Using the GUI, on page 43 or Importing a Function Profile Using the GUI, on page 44.

8 Create a service graph template and either use a function profile or enter the Layer 4 to Layer 7 parameters by hand.

See Creating a Layer 4 to Layer 7 Service Graph Template Using the GUI, on page 44.

9 Apply the service graph template.

See Applying a Service Graph Template to Endpoint Groups Using the GUI, on page 45.

10 Create the logical device context (optional if you used the GUI wizard).

See Creating a Device Selection Policy Using the GUI, on page 50.

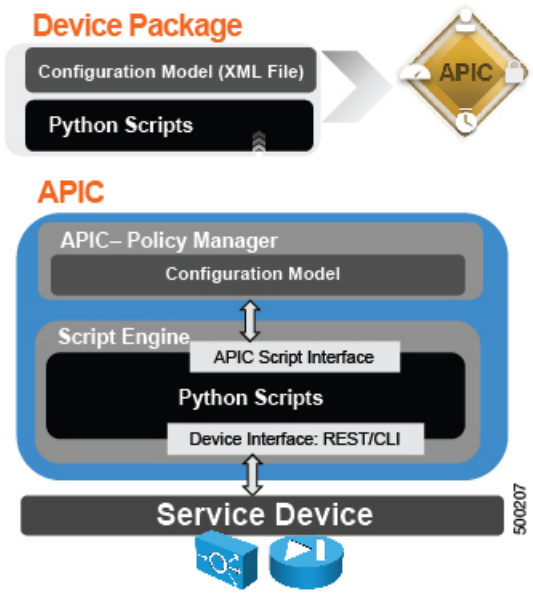
# About APIC-to-Layer 4 to Layer 7 Device Communication

The Application Policy Infrastructure Controller (APIC) requires a device package so that it can communicate with the Layer 4 to Layer 7 device. The device package performs the following functions to enable the communication:

- Service functions are added to the APIC through device package
- The device package contains a device model and device scripts (written in Python)
- The device model defines the service function and configuration
- Device scripts translate APIC API callouts to device-specific callouts

- Device scripts can interface with the device using REST or SSH

***Figure 31: Device Package and APIC***

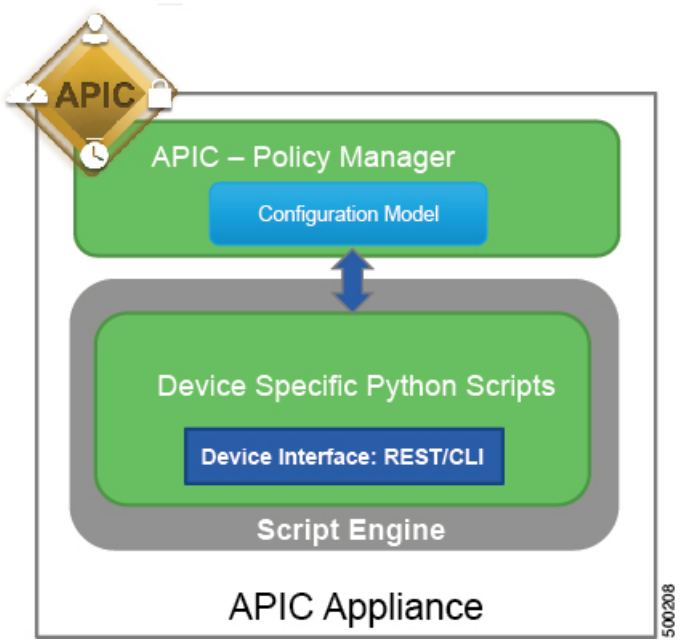



The functions in device script are classified into three categories:

- Device/infrastructure—For device-level configuration and monitoring
- Service events—For configuring functions, such as a server load balancer or SSL, on the device
- Endpoint or network events—For handling endpoint and network attach/detach events

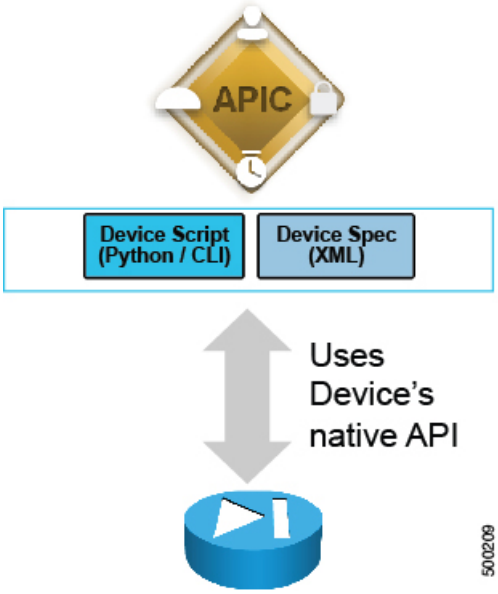
APIC uses the device configuration model provided in the package to pass the appropriate configuration to the device scripts. Device script handlers interface with the device using its REST interface or CLI.

***Figure 32: How the Device Scripts Interface with a Service Device***




The APIC interfaces with the device by using Python scripts. The APIC calls a device-specific python script function on various events.

***Figure 33: The APIC Interfaces with the Device Using Python Scripts***



The only configuration needed on the Layer 4 to Layer 7 device is management access. You can enable this access by enabling SSH, enabling HTTP access, and configuring the credentials on the device.

# About Layer 4 to Layer 7 Configuration Parameters

Each device package defines Layer 4 to Layer 7 configuration parameters, which configure the functions in a service graph. Parameters are always in key and value pairs. The `device_specification.xml` file within the device package defines the `vnsMDevCfg` object, which is the model configuration. The `vnsMDevCfg` object defines the Layer 4 to Layer 7 parameters.

For more information about configuration parameters, see the *Cisco APIC Layer 4 to Layer 7 Services Deployment Guide* .

You can use a function profile so that you can reuse the same parameter values when deploying a service graph. For more information, see About Function Profiles, on page 42.

# Setting Up Management Access to the Layer 4 to Layer 7 Device

The service graph configuration requires the Application Policy Infrastructure Controller (APIC) to communicate with the Layer 4 to Layer 7 device management address. For a physical appliance, the APIC communicates with the IP address of the management port of the appliance. In the case of a virtual appliance, the APIC communicates with the management address of the virtual appliance that is associated with the first vNIC (network adapter 1).

For out-of-band management, no particular configuration is needed on the APIC to establish this communication.

For in-band management, the APIC can talk to a physical or virtual appliance connected to the fabric through the Cisco Application Centric Infrastructure (ACI) fabric itself. You can configure a management endpoint group and create a pool of addresses that are used to apply Network Address Translation (NAT) to the IP address of the APIC to communicate with the appliance. You implement this capability by creating an endpoint group for management and a management address pool in this endpoint group.

The APIC is a clustered set of servers, each with its own IP address. You can configure the service graph from any of the controllers, and the configurations and the device package are replicated. Only one controller will talk to the Layer 4 to Layer 7 device to configure it, and you do not need to know which one it is except for troubleshooting purposes.

# Importing a Device Package Using the GUI

Before performing any configuration based on service graphs, you must download and install the appropriate device package in the Application Policy Infrastructure Controller (APIC) . A device package specifies to the APIC what devices you have and what the devices can do.

### Procedure

**Step 1** Download an appropriate device package. You can find the list of partners at the following URL: http://www.cisco.com/c/en/us/solutions/data-center-virtualization/ecosystem.html

This URL is the Partner Ecosystem page, where you can download the appropriate device package.

**Step 2** Log in to the APIC as the provider administrator.

**Step 3** On the menu bar, choose **L4-L7 Services** > **Packages**.

**Step 4** In the **Navigation** pane, choose **L4-L7 Service Device Types**.

**Step 5** In the **Work** pane, choose **Actions** > **Import Device Package**. The **Import Device Package** dialog box appears.

**Step 6** Click **Browse...** and browse to the device package that you want to use.
For information about creating device packages, see the *Cisco APIC Layer 4 to Layer 7 Device Package Development Guide*.

**Step 7** Click **Open**.

**Step 8** Click **Submit**.

# Creating Bridge Domains and VRFs Using the GUI

The data plane connectivity of the Layer 4 to Layer 7 devices is based on bridge domains. You need to create at least two bridge domains: one for the client side (or outside or consumer side) and one for the server side (inside or provider side).

### Procedure

**Step 1** On the menu bar, choose **Tenants** > **All Tenants**.

**Step 2** In the Work pane, double click the tenant's name.

**Step 3** In the Navigation pane, choose **Tenant** *tenant_name* > **Networking** > **Bridge Domains**.

**Step 4** In the Work pane, choose **Actions** > **Create Bridge Domain**.

**Step 5** In the **Create Bridge Domain** dialog box, fill in the fields as required, except as specified below:

a) In the **VRF** drop-down list, choose the VRF with which to associate the bridge domain.
Optionally, choose **Create VRF** to create a new VRF. In the **Create VRF** dialog box, fill in the fields as required and then click **Submit**.

b) For **L2 Uknown Unicast**, choose flooding and keep **ARP Flooding** enabled. Depending on the deployment mode, you could also enabled hardware-proxy, but the description of this configuration is outside of the scope of this guide.

c) For the **ARP Flooding** check box, put a check in the box if you will use ARP flooding. The most common configuration uses ARP flooding.

**Step 6** Click **Next**.

**Step 7** If you plan to provide the default gateway from the fabric bridge domain (typically the outside bridge domain), perform the following steps:

a) For the **Unicast Routing** check box, put a check in the box if you will use unicast routing.
If unicast routing is not enabled, the endpoints' IP addresses are not learned. If unicast routing is enabled, the endpoints' IP addresses are learned.

b) In the **Subnets** section, click + to add a subnet, and in the **Create Subnets** dialog box, fill in the fields as required.
Creating the subnets is required for creating a Layer 2 domain with routing.

Configure the default gateway by creating a subnet and entering an IP address.

If you plan to provide routing from the fabric, for example from the outside of the service graph, you must configure the subnet on the bridge domain.

**Step 8** Click **Next**.

**Step 9** Fill in the fields as required.

**Step 10** Click **Finish**.

# Creating Endpoint Groups and Contracts Using the GUI

The service graph requires a contract and subject to be associated with it. The service graph is deployed between a client-side, outside, consumer endpoint group and a server-side, inside, provider endpoint group. The first endpoint group is associated with the client-side or outside bridge domain and the second endpoint group is associated with the server-side or inside bridge domain.

### Procedure

**Step 1** On the menu bar, choose **Tenants** > **All Tenants**.

**Step 2** In the Work pane, double click the tenant's name.

**Step 3** In the Navigation pane, choose **Tenant** *tenant_name* > **Application Profiles**.

**Step 4** In the Work pane, choose **Actions** > **Create Application Profile**.

**Step 5** In the **Create Application Profile** dialog box, fill in the fields as required, except as specified below:

a) In the **EPGs** section, click + to create an endpoint group and fill in the fields as required.
In the **Provided Contract** and **Consumed Contract** fields, you can choose an existing contract, or you can choose **Create Contract** to create a new contract.

b) Click **Update**.

**Step 6** Click **Submit**.

# Logical Devices and Concrete Devices

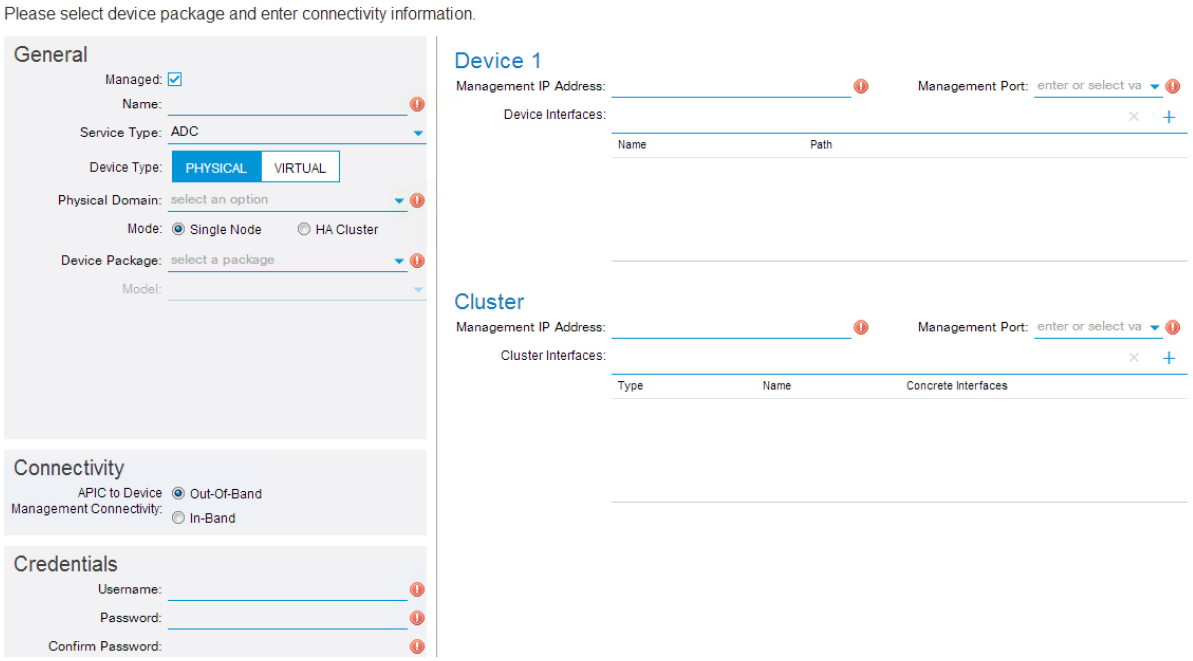
Cisco Application Centric Infrastructure (ACI) uses the following terminology:

- Concrete device—A concrete, or physical, device is a service device, such as a single load balancer or a single firewall.
- Logical device—A logical device is a cluster of two devices that operate in active-standby mode.
- Logical device context—The logical device context specifies the criteria for determining which specific device in the inventory to use to render a service graph.

Firewalls and load balancers are seldom deployed as single devices. Instead, they normally are deployed as clusters of active-standby pairs. Cisco ACI provides an abstraction to represent these clusters: the device cluster or logical device. The administrator must help ACI perform the mapping between the service graph and the clusters of firewalls and load balancers. The administrator also needs to tell ACI which pairs of concrete devices constitute a cluster. The GUI simplifies this process, guiding you through the steps to define each cluster of firewalls or load balancers.

The following screenshot shows the configuration dialog box for concrete and logical devices:

***Figure 34: Configuring Logical Devices Using the GUI***




The fields in this dialog box refer to the management information for the cluster of devices. The virtual address is the management address used when the pair of firewalls or load balancers is operating in active-standby mode.

# About Model Choice

When you create a concrete device (`CDev`), you can choose the model for a given device package, matching the model to the type of device that you are configuring. In some cases, you might want to choose the option `Unknown`, which is the generic model type. With this option, you have more control over the definition of the type of device and whether or not the device is context aware.

# About Connectivity Options

Under the concrete device definition, you must specify which domain to use. This setting allows Cisco Application Centric Infrastructure (ACI) to locate the device if it is on a virtualized server and provides a pool of VLANs that ACI can use to create the connectivity. Another option available as part of the concrete device

configuration is `EPG`. This option appears only if you are configuring in-band management and if this is the endpoint group that provides management access to the virtual appliance. In this case, vNIC1 on the virtual appliance is connected to this endpoint group.

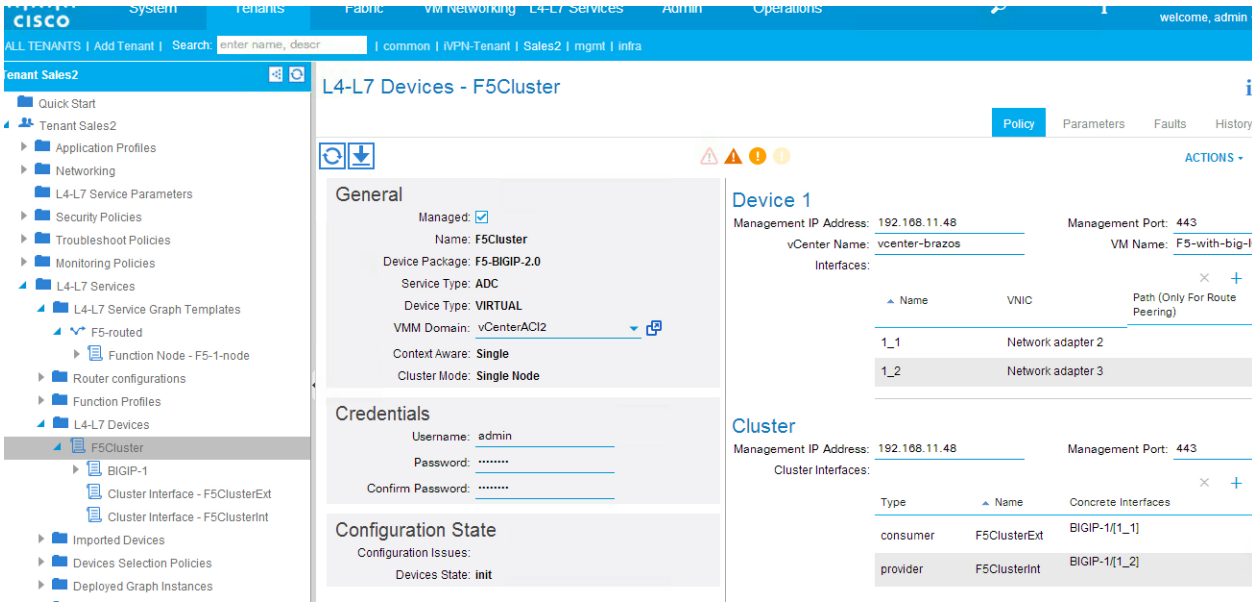
# About Interface Numbering

When using the GUI, you must configure logical interfaces. A logical interface defines a naming convention for the building block of the cluster and its mapping to the concrete device and to the metadevice.

For example, the metadevice of an F5 load balancer defines an external and an internal interface. The cluster model in Cisco Application Centric Infrastructure (ACI) defines two interfaces and lets you choose the name (logical interface, or `Lif`). Each interface maps to a metadevice interface and also to a physical (concrete) device interface. This process allows ACI to render the graph correctly.

The following screenshot shows an example of a mapping of a logical interface to a concrete device interface:

***Figure 35: Mapping of a Logical Interface to a Concrete Device Interface***




The interfaces have different names on the service device itself than the names that they have as part of the ACI configuration. For example, in the case of F5, the interfaces are numbered 1.1, 1.2, and so on. ACI allows you to reference these interfaces using the character "_" as a replacement for the "/" and "." characters. For example, F5 interfaces are referred to as 1_1, 1_2, and so on.

# Creating a Logical or Concrete Device Using the GUI

You can use a virtual or a physical Layer 4 to Layer 7 device in a service graph. You configure a concrete device to provide information to the Application Policy Infrastructure Controller (APIC) about where the device is and how to manage it. You configure a logical device to provide information to the APIC about the HA pair of Layer 4 to Layer 7 devices that Cisco Application Centric Infrastructure (ACI) can use for service graph purposes.

When you connecting to a physical device, you specify the physical interface. When you connect to a virtual machine, you specify the VMM domain, the virtual machine, and the virtual interfaces.

When defining a logical or concrete device, ACI gives you the option to choose which type of device it is by using the information included in the device package. Additionally, you can select an unknown model.

### Procedure

**Step 1** On the menu bar, choose **Tenants** > **All Tenants**.

**Step 2** In the Work pane, double click the tenant's name.

**Step 3** In the Navigation pane, choose **Tenant** *tenant_name* > **L4-L7 Services** > **L4-L7 Devices**.

**Step 4** In the Work pane, choose **Actions** > **Create L4-L7 Devices**.

**Step 5** In the **Create L4-L7 Devices** dialog box, fill in the fields as required, except as specified below:

a) Put a check in the **Managed** check box.

b) In the **Service Type** drop-down list, choose **ADC**, **Firewall**, or **IPS/IDS**.

c) In the **Physical Domain** or **VMM Domain** drop-down list, choose the domain to use.
For a concrete device, the domain allows ACI to locate the device if it is on a virtualized server and provides a pool of VLANs that ACI can use to create the connectivity.

d) In the **Model** drop-down list, choose the model that matches the type of device that you are configuring. For a concrete device, in some cases you might want to choose **Unknown** for the model, which is the generic model type. With this option, you have more control over the definition of the type of device and whether or not the device is context aware.

e) In the **Connectivity** section, in the **EPG** drop-down list, choose a management endpoint group. This field appears only if you are configuring in-band management and if this is the endpoint group that provides management access to the virtual appliance.

f) (Only for physical devices) In the **Device 1** section, fill in the fields as required, except as specified below:

1. In the **Management IP Address** field, enter the IP address used to manage the Layer 4 to Layer 7 device.
2. In the **Physical Interfaces** section, the **Name** must use a syntax that works on the Layer 4 to Layer 7 device.

g) (Only for virtual devices) In the **Device 1** section, fill in the fields as required, except as specified below:

1. In the **VM** drop-down list, choose the appropriate virtual machine.

h) (Only for physical devices) In the **Cluster** section, for **Cluster Interfaces**, for the **Type**, choose **consumer** for outside or client-side interfaces, and **provider** for inside or server-side interfaces.

**Step 6** Complete the remainder of the dialog screens and click **Finish** on the last screen.

**Step 7** Verify that the device is stable. In the Navigation pane, choose *tenant_name* > **L4-L7 Services** > **L4-L7 Devices** > *device_name*.

**Step 8** In the Work pane, in the **Configuration State** section, the device is stable if the **Device State** is `stable`.
If the device does not show as `stable`, verify the faults in the **Operations** tab and verify that the management address of the Layer 4 to Layer 7 device is reachable.

# Creating a Logical or Concrete Device with an HA Cluster Using the GUI

An HA configuration is set up at two levels: the logical device level, and the Layer 4 to Layer 7 parameters level.

At the logical device level, you tell Cisco Application Centric Infrastructure (ACI) which interfaces are the same interface. For example, the outside is made of 2 outside interfaces of each appliance. You also indicate which ports are failover link ports.

At the Layer 4 to Layer 7 parameters level, you provide the IP address for each interface. You might need to make some IP addresses "floating". You also provide the IP address for the failover links. For a physical ASA HA configuration, the IP address for the cluster interface is the admin context IP address.

### Procedure

**Step 1** On the menu bar, choose **Tenants** > **All Tenants**.

**Step 2** In the Work pane, double click the tenant's name.

**Step 3** In the Navigation pan**Tenants** > **All Tenants**e, choose **Tenant** ***tenant_name*** > **L4-L7 Services** > **L4-L7 Devices**.

**Step 4** In the Work pane, choose **Actions** > **Create L4-L7 Devices**.

**Step 5** In the **Create L4-L7 Devices** dialog box, fill in the fields as required, except as specified below:

a) Put a check in the **Managed** check box.

b) In the **Service Type** drop-down list, choose **ADC** or **Firewall**.

c) For the **Mode** radio buttons, click **HA Cluster**.

d) For the **Device 1** and **Device 2** sections, for **Device Interfaces**, configure the same failover LAN and failover link device interfaces for both devices.

e) (Only for physical devices) In the **Device 1** and **Device 2** sections, fill in the fields as required, except as specified below:

1. In the **Management IP Address** field, enter the IP address used to manage the Layer 4 to Layer 7 device.
2. In the **Physical Interfaces** section, the **Name** must use a syntax that works on the Layer 4 to Layer 7 device.

f) In the **Cluster** section, for **Cluster Interfaces**, for the **Type**, choose **consumer** for outside or client-side interfaces, and **provider** for inside or server-side interfaces.

**Step 6** Complete the remainder of the dialog screens and click **Finish** on the last screen.

**Step 7** Verify that the device is stable. In the Navigation pane, choose ***tenant_name*** > **L4-L7 Services** > **L4-L7 Devices** > ***device_name***.

**Step 8** In the Work pane, in the **Configuration State** section, the device is stable if the **Device State** is `stable`. If the device does not show as `stable`, verify the faults in the **Operations** tab and verify that the management address of the Layer 4 to Layer 7 device is reachable.

## Verifying the Status of a Logical or Concrete Device

You can verify the status of the device appliance by viewing the logical or concrete device configuration to see if it is stable.

### Procedure

**Step 1** On the menu bar, choose **Tenants** > **All Tenants**.

**Step 2** In the Work pane, double click the tenant's name.

**Step 3** In the Navigation pane, choose **Tenant** *tenant_name* > **L4-L7 Services** > **L4-L7 Devices** > *device_name*.

**Step 4** In the Work pane, choose the **Policy** tab.

**Step 5** Choose **Actions** > **Re-Query For Device Health**.

**Step 6** In the **Confirmation** dialog box, click **Yes**.

**Step 7** In the **Configuration State** section, ensure that the **Device State** of the device is `stable`.
If the device is stable, then it has been discovered and has connectivity with Cisco Application Centric Infrastructure (ACI).

# Function Profiles

## About Function Profiles

A function profile is a collection of pre-configured Layer 4 to Layer 7 parameters. Entering the Layer 4 to Layer 7 parameters is tedious and error prone due to the often large number of parameters that must be entered manually and individually. The function profile solves this problem since it is reusable. A function profile is specific to one node or function used in the template. The function profile is like an XML DTD with a reduced set of fields or with fields that are pre-populated based on the specific use case.

For example, a function profile for a web service could have the following preset parameters:

- L4 port = 80
- Load Balancing Method: Round Robin

Whenever you use this function profile, these parameters will automatically be set. You can then add more parameters or edit the preset parameters as necessary.

When you apply a service graph template, you can choose a function profile to deploy with the template. Function profiles can be also already part of the device package, in which case you only need to edit the function profiles to complete them when you deploy the service graph template.

When you create a function profile, some of the parameters are mandatory. Most of the mandatory parameters are highlighted in red in the GUI, but you must refer to the vendor's device package documentation to verify which parameters are mandatory.

# Creating a Function Profile Using the GUI

A Function Profile provides the default values for your service graph template. The following procedure explains how to create a new function profile.

### Procedure

**Step 1** On the menu bar, choose **Tenants** > **All Tenants**.

**Step 2** In the Work pane, double click the tenant's name.

**Step 3** In the Navigation pane, choose **Tenant** *tenant_name* > **L4-L7 Services** > **Function Profiles**.

**Step 4** Right click **Function Profiles** and choose **Create L4-L7 Services Function Profile**.

**Step 5** In the **Create L4-L7 Services Function Profile** dialog box, fill in the fields as required, except as specified below:

a) In the **Profile Group** drop-down list, choose **Create Function Profile Group**.
A profile group is a mechanism that allows you to group your profiles together for organizational purposes. For example, you may want to create a profile for your Web, legacy, or e-mail applications. You can create groups and then you can put your profiles into those groups. You may see that you already have an existing group available, but if you do not, then you can create a new one by naming it and providing a description in the **Create L4-L7 Services Function Profile Group** window.

**Step 6** In the **Create L4-L7 Services Function Profile Group** dialog box, fill in the fields as required.

**Step 7** Click **Submit**.
Now you have successfully completed and saved a profile group, which now appears in the **Create L4-L7 Services Function Profile** dialog box.

A profile is created for a particular service function. What you choose from the **Device Function** drop-down list in the **Create L4-L7 Services Function Profile** is the function for which you are writing a profile. From the drop-down list, you will see a list of device packages with service functions available in the Application Policy Infrastructure Controller (APIC) after you have imported the device packages.

**Step 8** Back in the **Create L4-L7 Services Function Profile** dialog box, remove the check from the **Copy Existing Profile Parameters** check box.

**Step 9** In the **Device Function** drop-down list, choose a device package that has a function.
Options are displayed with the various parameters that are part of that function. The purpose of the profile is to provide the default values for the parameters.

**Note** At this point none of these parameters have any values, but you can add them, which are then used as the default values. The function profiles can be used by the graph templates after you provide these values. These values are applied to the graph template as default values, which means that if you use the graph templates and you do not provide a value for that particular parameter, then the APIC looks up the profile and see if the value is there. If it is there, then the APIC uses that.

**Step 10** Add values in the **Features and Parameters** section at the bottom of the **Create L4-L7 Services Function Profile** window. There are two tabs, **Basic Parameters** and **All Parameters**. The **Basic Parameters** tab includes a list of parameters that are marked as mandatory (required) in the package. The **All Parameters** tab includes a list of the basic parameters as well as some additional / optional ones for advanced configurations. The reason we expose the **Basic Parameters** is because these are part of the basic configuration and the administrator is expected to fill these out. **All Parameters** are optional so unless you want to customize the functionality, these parameters can be left out.

**Step 11** Click **Submit**.

Now you have completed and saved your function profile.

## Importing a Function Profile Using the GUI

If you already have a function profile that you saved to your local machine, you can import it into (post it to) the Application Policy Infrastructure Controller (APIC).

### Procedure

**Step 1** On the menu bar, choose **Tenants** > **All Tenants**.

**Step 2** In the Work pane, double click the tenant's name.

**Step 3** In the Navigation pane, choose ***tenant_name*** > **L4-L7 Services** > **Function Profiles** > ***function_profile_group_name***.
*function_profile_group_name* is the function profile group into which you want to import the function profile.

**Step 4** Right click *function_profile_group_name* and choose **Post ...**.

**Step 5** In the **Post** dialog box, click **Browse...** and browse to the function profile's XML or JSON file.

**Step 6** Click **Post**.

**Step 7** (Optional) Modify or add to the parameters in the imported function profile.

# Service Graph Templates

## Creating a Layer 4 to Layer 7 Service Graph Template Using the GUI

A service graph template is a sequence of Layer 4 to Layer 7 functions or devices and their associated configuration, which can be provided by using function profiles. The service graph template must be associated with a contract to be "rendered"—or configured—on the Layer 4 to Layer 7 device and on the fabric.

### Before You Begin

- You must have configured a tenant.

### Procedure

**Step 1** On the menu bar, choose **Tenants** > **All Tenants**.

**Step 2** In the Work pane, double click the tenant's name.

**Step 3** In the Navigation pane, choose **Tenant** *tenant_name* > **L4-L7 Services** > **L4-L7 Service Graph Templates**.

**Step 4** In the Work pane, choose **Actions** > **Create a L4-L7 Service Graph Template**.

**Step 5** In the **Create a L4-L7 Service Graph Template** dialog box, in the **Device Clusters** section, choose a device cluster.

**Step 6** Complete the following fields:

| Name | Description |
|---|---|
| **Graph Name** field | Enter the name of the service graph template. |
| **Graph Type** radio buttons | Choose to create a new service graph template or clone an existing service graph template. |
| **Existing Graphs** drop-down list | (Only for cloning an existing service graph template) Choose an existing service graph template to clone. |

**Step 7** (Only for creating a new service graph template) Drag a device from the **Device Clusters** section and drop it between the consumer endpoint group and provider endpoint group to create a service node.

**Step 8** (Optional) (Only for cloning an existing service graph template) Remove the existing node and drag a different device cluster to the node area to create a service node.

**Step 9** Click **Submit**.

**Step 10** (Optional) In the **Navigation** pane. click the service graph template.
The screen presents a graphic topology of the service graph template.

# Applying a Service Graph Template to Endpoint Groups Using the GUI

The following procedure explains how to apply a service graph template to endpoint groups:

### Before You Begin

You must have created the following things:

- Application endpoint groups
- A service graph template

### Procedure

**Step 1** On the menu bar, choose **Tenants** > **All Tenants**.

**Step 2** In the Work pane, double click the tenant's name.

**Step 3** In the Navigation pane, choose **Tenant** ***tenant_name*** > **L4-L7 Services** > **L4-L7 Service Graph Templates** > ***template_name***.

**Step 4** In the Work pane, choose **Actions** > **Apply L4-L7 Service Graph Template**.
You will be associating a Layer 4 to Layer 7 service graph template to your consumer and provider endpoint groups.

**Step 5** In the **Apply L4-L7 Service Graph Template To EPGs** dialog, in the **EPG Information** section, complete the following fields:

| Name | Description |
|---|---|
| **Consumer EPG/External Network** drop-down list | Choose a consumer endpoint group. |
| **Provider EPG/External Network** drop-down list | Choose a provider endpoint group. |

**Step 6** In the **Contract Information** section, complete the following fields:

| Name | Description |
|---|---|
| **Contract** radio buttons | Choose to create a contract or choose an existing contract. |
| **Contract Name** field | (Only for creating a contract) Enter the name of the contract. |
| **No Filter (Allow All Traffic)** check box | (Only for creating a contract) Put a check in the box to allow all traffic, or remove the check from the box to filter traffic. |
| **Filter Entries** | (Only for filtering traffic) Click + and enter the filter information, then click **Update**. |
| **Existing Contract With Subjects** drop-down list | (Only for choosing an existing contract) Choose an existing contract. |

**Step 7** Click **Next**.

**Step 8** In the **Device Clusters** section, choose a device cluster.

**Step 9** Complete the following field:

| Name | Description |
|---|---|
| **Graph Template** drop-down list | Choose a graph template. |

**Step 10** (Optional) Remove the existing node and drag a different device cluster to the node area to create a service node.

**Step 11** In the **unmanaged information** section, complete the following field:

| Name | Description |
|---|---|
| **Cluster Interface For Consumer Connector** drop-down list | Choose an interface for the consumer connector. |
| **Cluster Interface For Provider Connector** drop-down list | Choose an interface for the provider connector. |
| **General** check box | Put a check in the box to be able to choose bridge domains. |
| **BD For Consumer Connector** drop-down list | (Only if you put a check in the **General** check box) Choose a bridge domain for the consumer connector. The bridge domain is used for the data path traffic. |
| **BD For Provider Connector** drop-down list | (Only if you put a check in the **General** check box) Choose a bridge domain for the provider connector. The bridge domain is used for the data path traffic. |
| **Route Peering** check box | Put a check in the box to enable route peering. |

The Application Policy Infrastructure Controller (APIC) uses the chosen bridge domains for data path traffic between function nodes as required by the chosen service graph template. Refer to the online help for the service graph templates to learn more about how this bridge domain is used.

**Step 12** (Only for managed devices) Click **Next**.

**Step 13** (Only for managed devices) In the **Parameters** screen, in the **Required Parameters** tab, enter the names and values, as appropriate, for all of the required parameters.

**Step 14** Click **Finish**.
You now have an active service graph template. The APIC populates the Layer 4 to Layer 7 parameters based on the chosen function profile and colors the mandatory parameters in green if they are configured correctly.

# Verifying a Service Graph Deployment Using the GUI

After you apply a service graph template, the service graph is associated with a contract and you can see the list of deployed graphs and rendered concrete devices in the Layer 4 to Layer 7 devices portion of the GUI. When the graph is rendered, you will see configurations appear in the device that is part of the graph. The following procedure verifies that the service graph deployed successfully.

## Procedure

**Step 1** On the menu bar, choose **Tenants** > **All Tenants**.

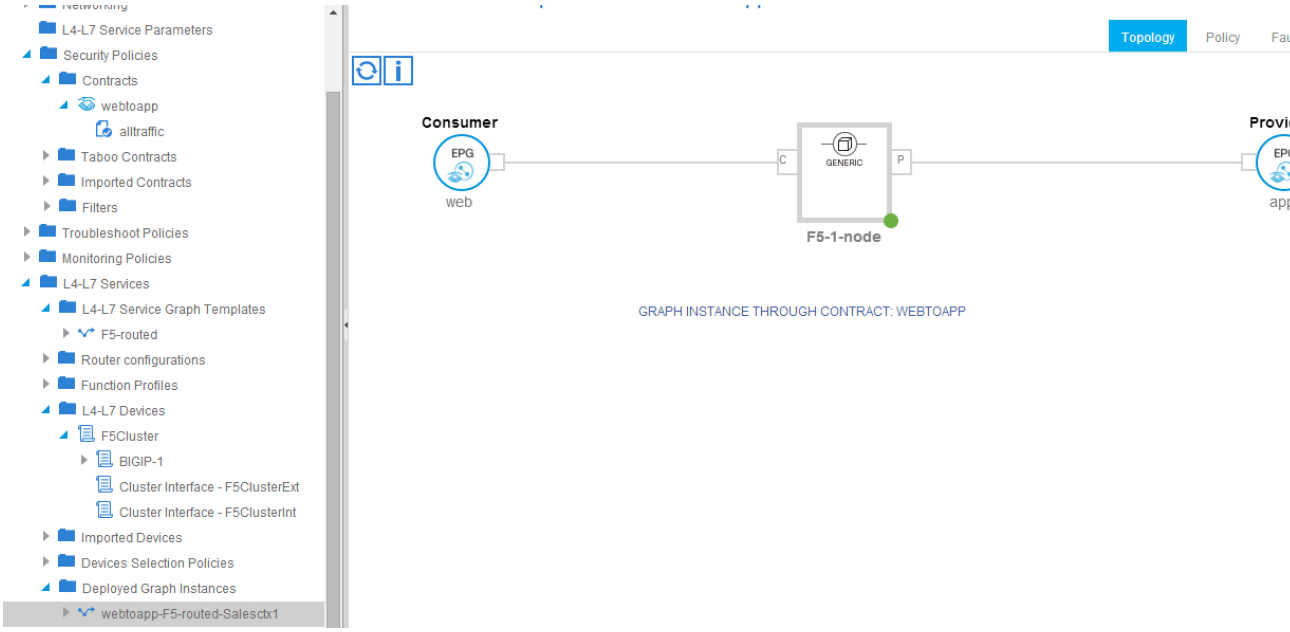
**Step 2** In the Work pane, double click the tenant's name.

**Step 3** In the Navigation pane, choose **Tenant** *tenant_name* > **L4-L7 Services** > **Deployed Devices**.

**Step 4** In the Work pane, look for the device in the table. If the service graph deployment failed, you will not see the device.

**Step 5** In the Navigation pane, choose **Tenant** *tenant_name* > **L4-L7 Services** > **Deployed Graph Instances** > ***graph_name***.
The following screenshot shows an example of deployed (rendered) service graphs:

***Figure 36: Deployed Service Graphs***

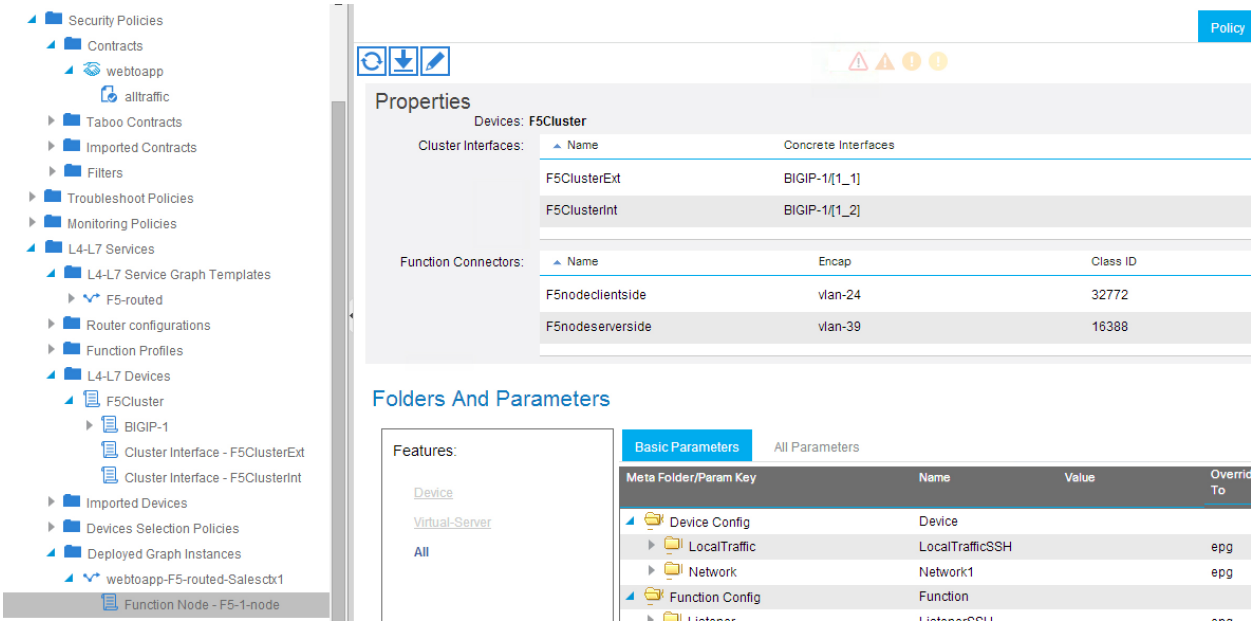



If the service graph has been deployed, the graph is listed in the **Deployed Graph Instances** folder.

a) (Optional) View which configurations were not applied and why by choosing **Faults** tab.

**Step 6** In the Navigation pane, choose **Tenant** ***tenant_name*** > **L4-L7 Services** > **Deployed Graph Instances** > ***graph_name*** > ***function_node_name*** to see which VLANs and which port groups have been allocated by Cisco Application Centric Infrastructure (ACI) to establish the connectivity with the service device.

***Figure 37: Allocated VLANs and Port Groups***




In the case of virtual appliances, ACI creates some port groups called shadow endpoint groups, and ACI moves the vNIC of the appliance to these port groups.

# Undoing a Service Graph Configuration Using the GUI

If you no longer need a service graph configuration, then you can delete the service graph template.

### Procedure

**Step 1** On the menu bar, choose **Tenants** > **All Tenants**.

**Step 2** In the Work pane, double click the tenant's name.

**Step 3** In the Navigation pane, choose **Tenant** ***tenant_name*** > **L4-L7 Services** > **L4-L7 Service Graph Templates** > ***service_graph_template_name***.

**Step 4** Right click the service graph template and choose **Remove Related Objects Of Graph Template**.

**Step 5** Right click the service graph template and choose **Delete**.

# Creating a Device Selection Policy Using the GUI

If you did not use the **Apply L4-L7 Service Graph Template To EPGs** wizard to apply the service graph template, you might need to configure a device selection policy (also known as a logical device context). The device selection policy instructs Cisco Application Centric Infrastructure (ACI) about which firewall or load balancer device to use to render a graph.

If you used the **Apply L4-L7 Service Graph Template To EPGs** wizard to apply the service graph template, then a device selection policy was configured automatically and you do not need to configure one manually.

**Note** When using the NX-OS-style CLI, the device selection policy is configured automatically; there are no equivalent NX-OS-style CLI commands.

### Procedure

**Step 1** On the menu bar, choose **Tenants** > **All Tenants**.

**Step 2** In the Work pane, double click the tenant's name.

**Step 3** In the Navigation pane, choose **Tenant** *tenant_name* > **L4-L7 Services** > **Devices Selection Policies**.

**Step 4** In the Work pane, choose **Actions** > **Create Logical Device Context**.

**Step 5** In the **Create Logical Device Context** dialog box, fill in the fields as required, except as specified below:

a) In the **Service Type** drop-down list, choose the contract for the device selection policy. If you do not want to use the contract name as part of the criteria for using a device, choose **any**.

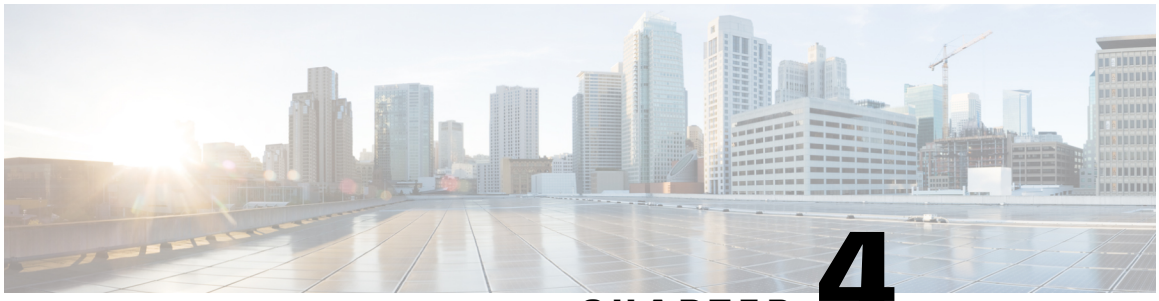
b) In the **Graph Name** drop-down list, choose the graph for the device selection policy. If you do not want to use the graph name as part of the criteria for using a device, choose **any**.

c) In the **Node Name** drop-down list, choose the node for the device selection policy. If you do not want to use the node name as part of the criteria for using a device, choose **any**.

**Step 6** In the **Cluster Interface Contexts** section, click + to add a cluster interface context.

- **Connector Name—**The name of the connector in the service graph template.
- **Logical Interface—**The logical interface to use for the connector that is specified in the logical interface context.
- **Bridge Domain—**The bridge domain that is specified in the logical interface context.
- **Subnets—**The subnet to configure on the logical interfaces when the service graph template is instantiated.

**Step 7** Click **Submit**.



CHAPTER 4

# Deploying F5

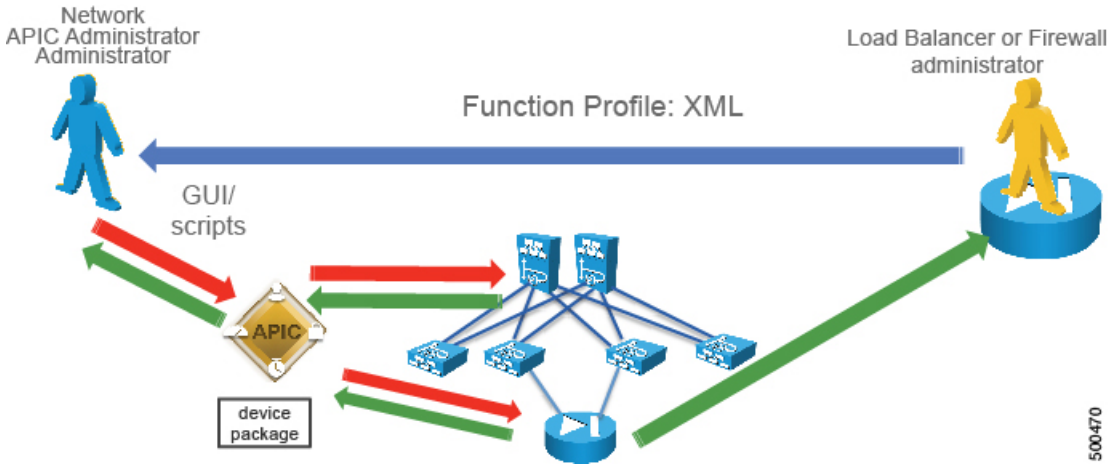


## About the F5 Operational Model

There are two main operational models that you can use with F5. In the first model, which is the default mode, the F5 configuration is managed through the Application Policy Infrastructure Controller (APIC). In the second model, the F5 administrator uses Big-IQ to define the Layer 4 to Layer 7 service configurations and the APIC administrator just associates them with the service graph. This document covers only the first deployment model.

The default operational model when using F5 with the APIC requires that all changes made to the F5 device are performed through the APIC. The following figure illustrates that the configuration of the F5 device is managed by the APIC:

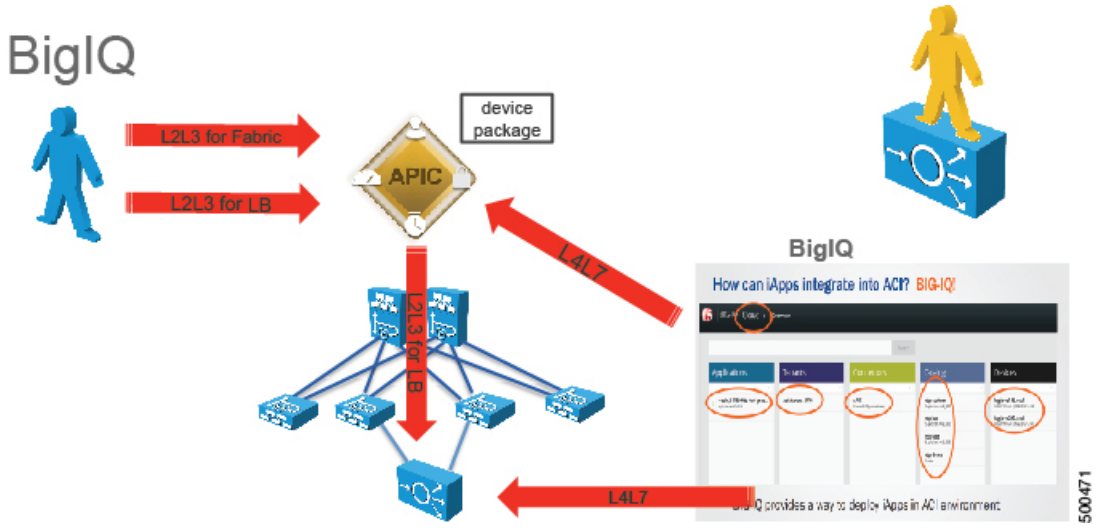
***Figure 38: Layer 4 to Layer 7 Service Administration with the APIC***




The F5 administrator provides the XML or JSON function profile configuration to the APIC administrator who then pushes the function profile through the APIC to the F5 device.

The following figure illustrates the management model whereby the F5 administrator defines iApps on Big-IQ, and the Layer 4 to Layer 7 parameters are passed to the APIC to be instantiated of the F5 appliance:

***Figure 39: Integrating iApps into ACI***




# Translation of F5 Terminology

The following table translates F5 terminology into Cisco load balancer terminology:

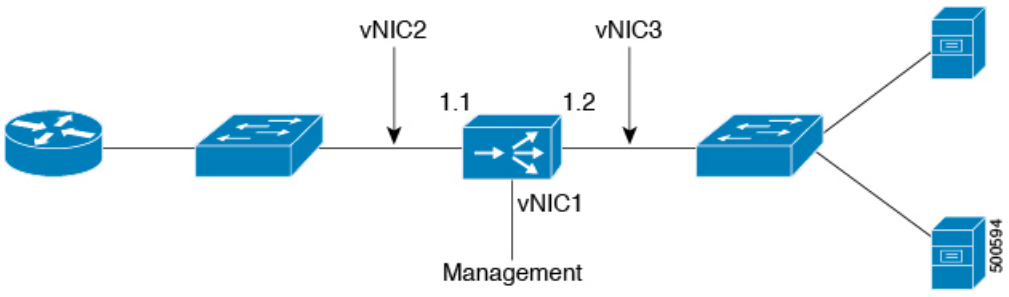
| F5 Terminology | Cisco Load Balancer Terminology |
|---|---|
| Listener IP | Virtual IP |
| Pool | Serverfarm |
| Pool Member | Real Server |
| SelfIP Address | Interface Address (alias in content switching module terminology) |
| Floating yes/no | Makes the SelfIP floating, such as HSRP or VRRP |
| Route Domain | VRF, normally one route domain per partition on F5 |

The following table translates which vNIC corresponds to which interface in F5 and Cisco Application Centric Infrastructure (ACI):

| Interface | VMware | F5 | ACI | IP address is entered as |
|---|---|---|---|---|
| Management | vNIC1 | Management | N/A | N/A |
| Outside | vNIC2 | 1.1 | 1_1 | ExternalSelfIP |
| Inside | vNIC3 | 1.2 | 1_2 | InternalSelfIP |

The following figure illustrates the naming convention for the interfaces in the case of an F5 load balancer:

*Figure 40: F5 Load Balancer Interface Naming Convention*




# About F5 Partitions

When you define the Layer 4 to Layer 7 device as multicontext, you can put the device into a tenant such as tenant `Common` and then export the device to multiple tenants. Multicontext support requires a physical appliance. The virtual appliance also supports multicontext in the sense that you can create multiple partitions, but a

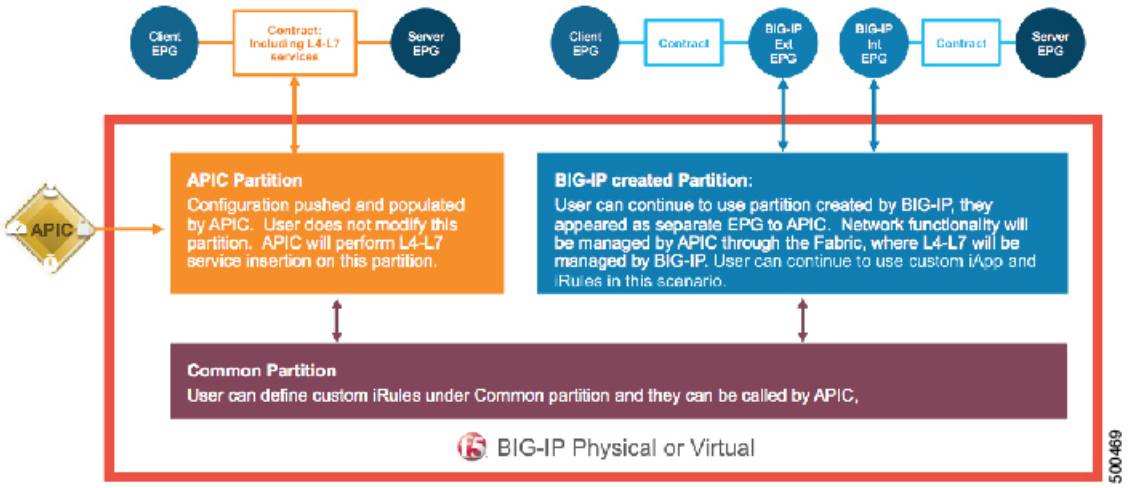
virtual appliance will not forward the traffic. Since the virtual appliance is on multiple tenants, the vNICs cannot be shared because there cannot be a trunk with VLANs on the same vNIC.

Application Policy Infrastructure Controller (APIC) uses tenant `Common` for objects that can be used by other tenants. For example, APIC has filters in the common partition that can be used from other tenants. F5 has a "common" partition that functions similar to tenant `Common`. The common partition has configurations to manage the F5 device. The configurations can be exported to other partitions, such as the "monitor" configurations, which correspond to "probes" or "keepalive" in Cisco load balancer terminology. APIC logs into the common partition, but creates a new F5 partition for each tenant.

You can have multiple virtual servers for different applications in the same BIG-IP partition or APIC tenant. A partition created by APIC inside BIG-IP is prefixed by "apic_", followed by the tenant ID to represent the partition in F5. For example, "apic_5437". The tenant ID is based on the service graph virtual device (VDev) ID. Each partition is assigned an individual route domain for Layer 3 separation. A virtual server created by APIC inside BIG-IP is prefixed by "apic_*tenant-ID*", followed by the service graph ID. For example, " apic_5437_3456".

When the APIC manages an F5 partition, the F5 administrator cannot make changes to that partition directly from the F5 interface. All changes must be performed exclusively using the APIC. A single F5 device can have partitions that are managed by the APIC and other partitions that are managed by the F5 administrator directly. This design approach is called "mixed mode". The following figure illustrates this concept:

***Figure 41: Mixed Mode Support***

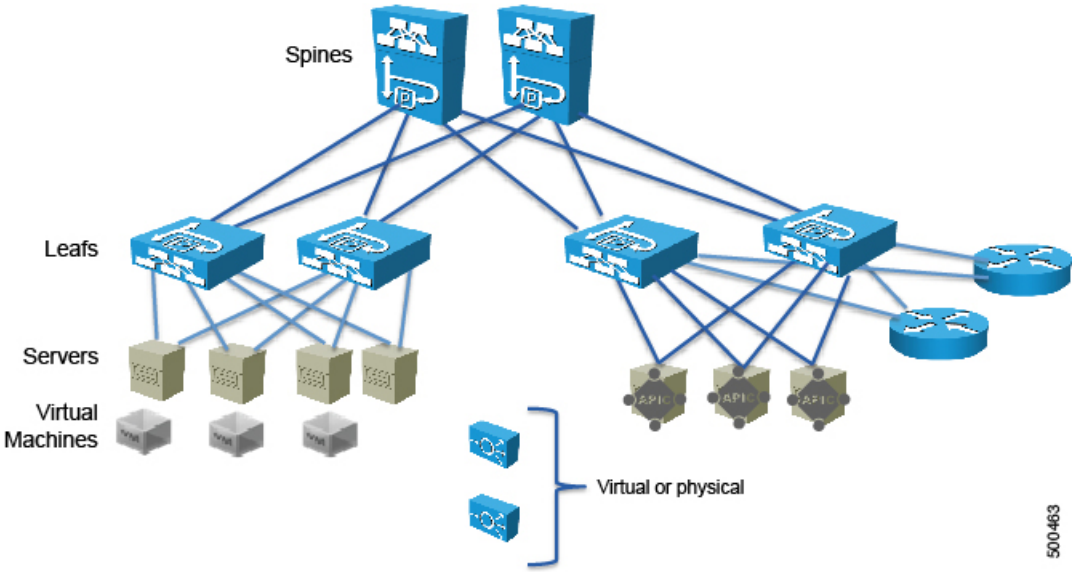


# F5 in GoTo Mode

## About Deploying F5 in GoTo Mode

The following figure illustrates the topology for deploying Cisco Application Centric Infrastructure (ACI) fabric with F5 devices:

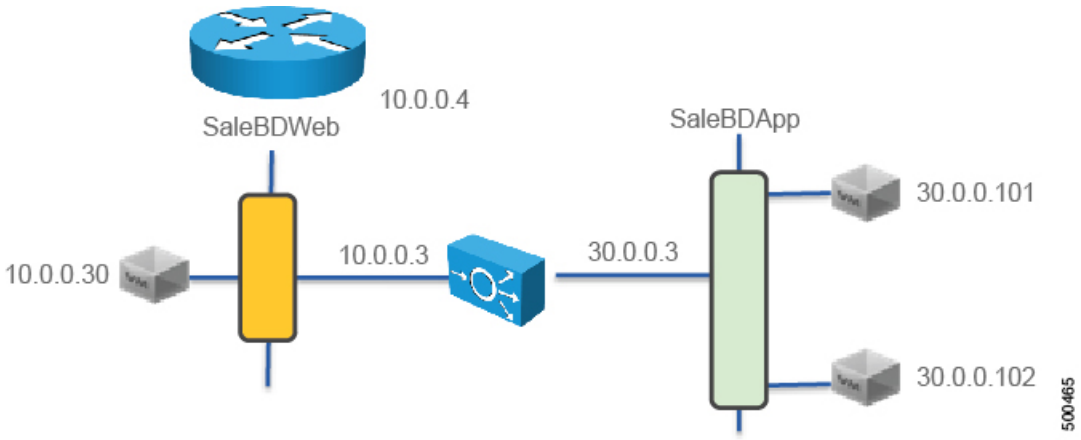
***Figure 42: ACI Fabric with F5 Devices***




The F5 load balancer can be connected as a physical or virtual device to any of the leafs in the topology.

The following figure illustrates the logical topology for an F5 GoTo deployment:

***Figure 43: Logical Topology for an F5 GoTo Deployment***



The F5 device is deployed as part of a contract connecting the EPG SaleBDWeb and SaleBDApp which in the picture are on subnets 10.0.0.x and 30.0.0.x respectively

To deploy an F5 device in the GoTo mode, you must perform the following steps:

1. Configure 2 bridge domains
2. Configure 2 endpoint groups, with each one associated with a different bridge domain
3. Configure the F5 device as a GoTo device
4. Configure a VIP on the same subnet as the bridge domain that the F5 connects to on the client side (outside, or consumer side)
5. Configure the contract between the outside and inside endpoint group (or server side or provider side)
6. Associate the service graph with the contract

   You must configure a different service graph instance for each virtual IP address.
7. Associate the external logical interface with 1_1 (which in the case of F5 VE is Network Adapter 2)
8. Associate the internal logical interface with 1_2 (which in the case of F5 VE is Network Adapter 3)

# Overview of Preparing an F5 Device in GoTo Mode

The following procedure provides an overview of preparing an F5 device to be deployed in GoTo mode.

### Procedure

**Step 1** In the APIC, define the VLAN pool to use.
**Step 2** If you are using a virtual appliance, create a virtual domain.
**Step 3** If you are using a physical appliance, create a physical domain.
**Step 4** Create the attach entity profile.
**Step 5** Download the device package from the F5 Web site.
**Step 6** Upload the device package to the APIC.

# Configuring Bridge Domains for F5 in GoTo Mode

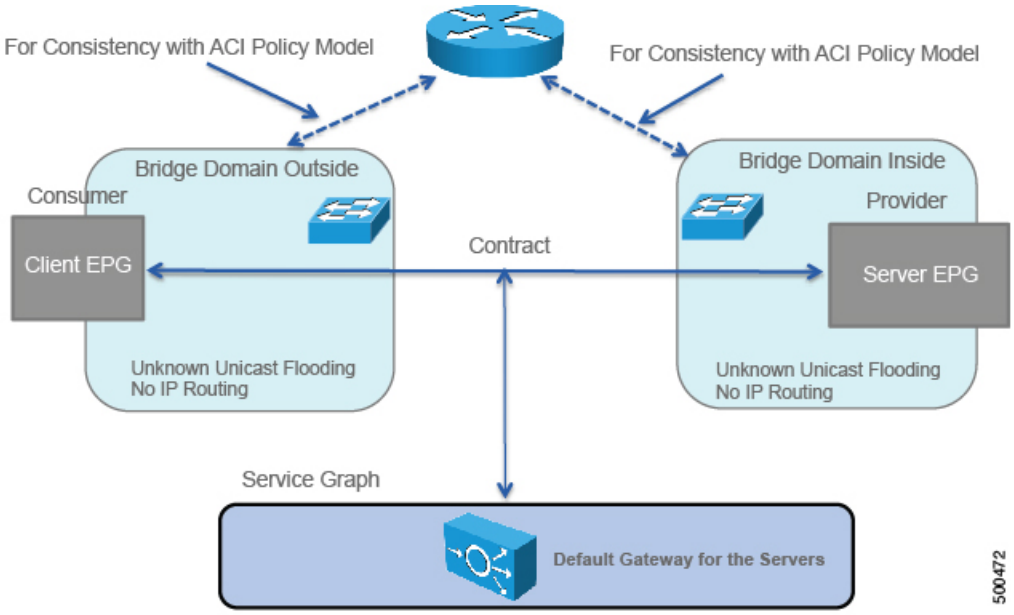
When you configure the bridge domains for F5 in GoTo mode, configure the bridge domains as you would for a generic configuration, except as follows:

- **L2 Unknown Unicast** radio buttons—Choose **Flood**.
- **ARP Flooding** check box—Put a check in the check box.
- **Unicast Routing** check box—This configuration depends on whether this is the outside bridge domain and whether you need the Cisco Application Centric Infrastructure (ACI) fabric to route. If you do not know, leave this check box unchecked.

For information on how to configure bridge domains, see Creating Bridge Domains and VRFs Using the GUI, on page 36.

The following figure illustrates a GoTo mode deployment without hardware-proxy and without endpoint attach:

***Figure 44: GoTo Mode Deployment Without Hardware Proxy and Without Endpoint Attach***

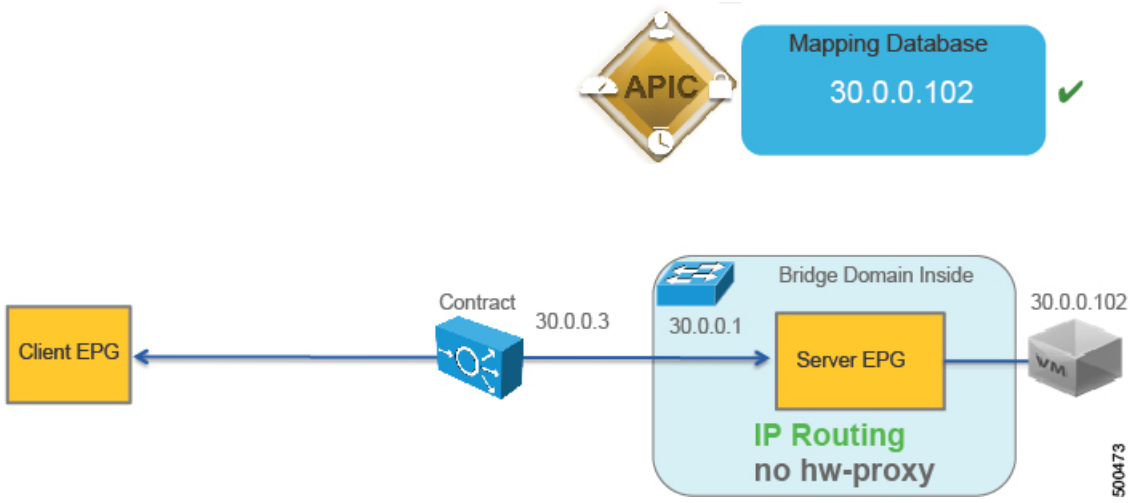



If you need the mapping database to learn the endpoints' IP addresses, you must enable unicast routing in the bridge domains.

# Adding Endpoint Attach Support for F5 in GoTo Mode

You can deploy an F5 device in a service graph in a way that the endpoints that are discovered in the provider endpoint group are automatically added to the pool of load balanced servers. In the F5 device, this feature is called "endpoint attach".

***Figure 45: How the Bridge Domain Inside the APIC with Routing Enabled Learns the IP address of the Endpoints***




The following procedure enables endpoint attach.

### Procedure

**Step 1** On the menu bar, choose **Tenants** > **All Tenants**.

**Step 2** In the Work pane, double click the tenant's name.

**Step 3** In the Navigation pane, choose **Tenant** *tenant_name* > **L4-L7 Services** > **L4-L7 Service Graph Templates** > *service_graph_template_name* > **Function Node -** *node_name* > **provider** or **Tenant** *tenant_name* > **L4-L7 Services** > **L4-L7 Service Graph Templates** > *service_graph_template_name* > **Function Node -** *node_name* > **internal**, as appropriate for the provider-side connector.

**Step 4** In the Work pane, choose the connector's properties.

**Step 5** Put a check in the **Attachment Notification** check box.

**Step 6** Click **Submit**.

The configuration in XML format is as follows:

```
<vnsAbsNode funcType="GoTo" name="F5-1-node" shareEncap="no" managed="yes">
<!-- This is specifies which function this is -->
    <vnsRsNodeToMFunc tDn="uni/infra/mDev-F5-BIGIP-2.0/mFunc-Virtual-Server"/>
    ...

<!-- This is the name of the connectivity point of the node -->
<!-- the name is referenced by "AbsNode-F5-1-node/AbsFConn-F5nodeserverside" -->
<!-- Attachment Notify is used to create a Pool of servers dynamically on the load balancer
 -->
```

```
    <vnsAbsFuncConn attNotify="yes" name="F5nodeserverside" >
<!-- This is the Metadevice information i.e. the mConnector -->
<!-- "internal" is not an arbitrary name, it is the definition of the type of interface -->
<!-- and it has a precise meaning in the meta device -->
    <vnsRsMConnAtt tDn="uni/infra/mDev-F5-BIGIP-2.0/mFunc-Virtual-Server/mConn-internal"/>
</vnsAbsFuncConn>
```

With these configurations in place, you can define the Layer 4 to Layer 7 parameters for the load balancer that refer to these endpoints. This is achieved in the Layer 4 to Layer 7 parameters as follows:

```
<vnsFolderInst  ctrctNameOrLbl="webtoapp" graphNameOrLbl="F5-routed" key="Pool" locked="no"
 name="ServerPoolSSH" nodeNameOrLbl="F5-1-node">
<!-- CONFIGURE LOAD BALANCING TYPE HERE -->
    <vnsParamInst  key="LBMethod" locked="no"  name="LBMethod"  value="ROUND_ROBIN"/>
<!-- Use Dynamic only if you want to use the EPG endpoints to autopopulate the serverfarm
pool -->
    <vnsParamInst key="PoolType" name="PoolType" value="DYNAMIC"/>
```

# Tuning the Server-Side Bridge Domain for Flood Removal for F5 in GoTo Mode

On the server-side bridge domain, it can be beneficial to reduce flooding for unknown unicast packets. To do this, you can enable hardware proxy on the bridge domain. You should keep ARP flooding enabled because it might be necessary in the presence of F5 deployed in HA pairs.

# F5 GoTo Mode Design Examples

The following figures illustrate F5 GoTo mode deployments with various scenarios: some with the client connected directly to the fabric, some with the fabric providing routing to the outside, and some with an external router. The figures include the recommended bridge domain settings for both client and server-side bridge domains.

The settings for the server-side or provider-side (also known as the internal bridge domain, `BD2`) include IP routing in case you decide to use the endpoint attach feature. If you do not want to use endpoint attach and

you do not care about flood reduction in the server-side bridge domain, you can configure the bridge domain without IP routing.

***Figure 46: GoTo Mode Deployment with Client Virtual Machines and a Split VRF***

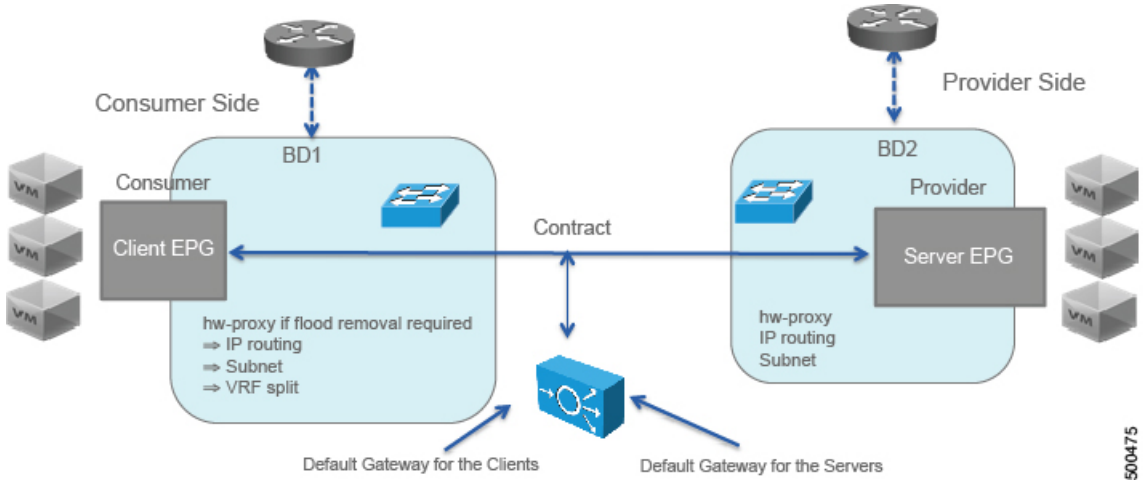



***Figure 47: GoTo Mode Deployment with a Layer 3 Outside and a Split VRF***

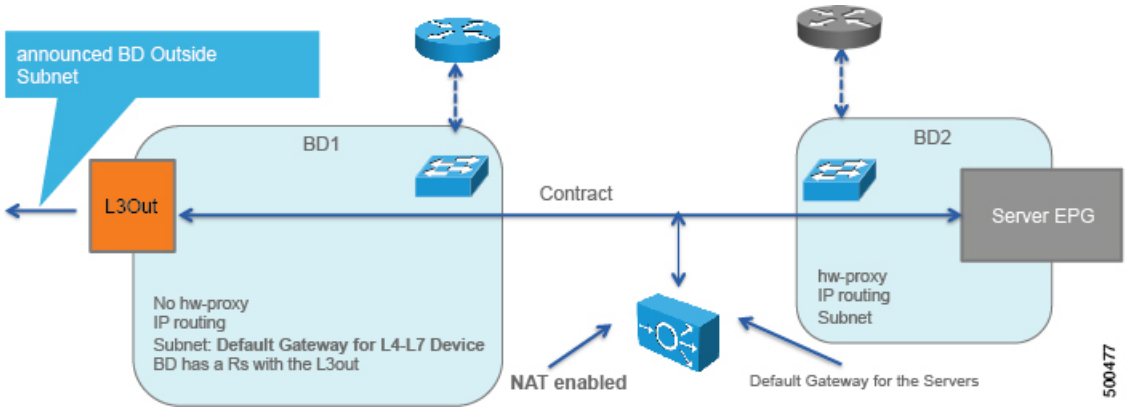
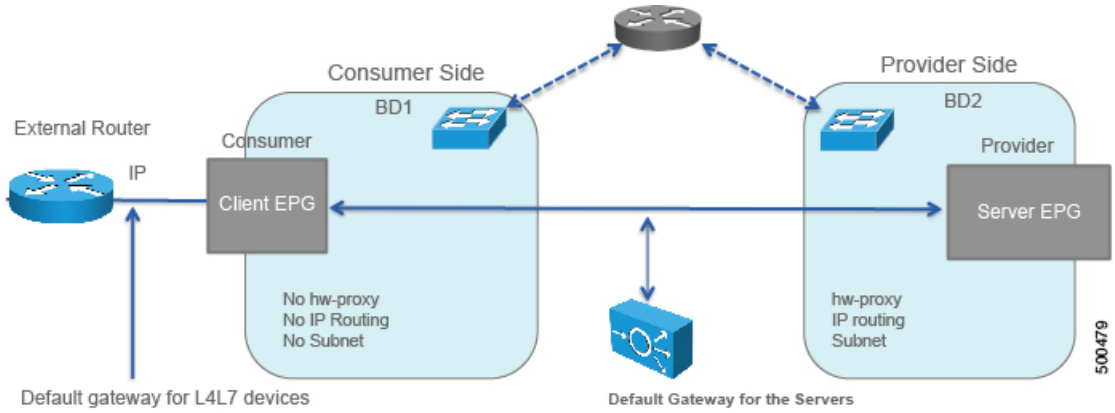



***Figure 48: GoTo Mode Deployment with an External Router***




# Deploying F5 in GoTo Mode

The tasks that you must perform to deploy F5 in GoTo mode are nearly identical to the tasks for generically deploying a service graph, with a few differences. The following procedure provides the generic service graph deployment tasks, along with information about what you must do differently to deploy F5 in GoTo mode.

### Procedure

**Step 1** Import the device package.
See Importing a Device Package Using the GUI, on page 35.

**Step 2** Create the bridge domains and VRFs.
See Creating Bridge Domains and VRFs Using the GUI, on page 36.

a) For the inside bridge domain, enable **Unicast Routing** if you plan to use endpoint attach.
b) Associate the bridge domain with a VRF, which is necessary because of the object model. The hardware will not program the VRF if the bridge domain is configured only as Layer 2.

**Step 3** Create endpoint groups and contracts.
See Creating Endpoint Groups and Contracts Using the GUI, on page 37.

**Step 4** Configure logical devices and concrete devices.
See Creating a Logical or Concrete Device Using the GUI, on page 39.

a) For a concrete device, in the **Service Type** drop-down list, choose **ADC** for a load balancer.
b) If the device is virtual, in the **VMM Domain** drop-down list, choose the appropriate VMM domain.
c) In the **Model** drop-down list, choose **BIG-IP-VE-GENERIC** for an F5 VE.
d) For the Layer 4 to Layer 7 parameters, for the Host Name parameter, set the value to the host name of the F5 device.

**Step 5** Create or import a function profile.
See Creating a Function Profile Using the GUI, on page 43 or Importing a Function Profile Using the GUI, on page 44.

- The Layer 4 to Layer 7 parameters under the `CDev` object refer to the common partition of F5; you do not need to use these parameters

- All the parameters with "-Default" in the value must be changed to something else, such as "Pool" or "Listener"
- If you use HA, you cannot use the GUI to configure the parameters because the SelfIP parameter value will be the same on the active and on the standby appliance

The following table describes the mandatory Layer 4 to Layer 7 parameters and provides examples of possible values that you must change for your specific configuration:

***Table 1: Layer 4 to Layer 7 Parameters for F5 in GoTo Mode***

| L4-L7 Parameter or Folder | Usage and Notes |
|---|---|
| `Listener IP` folder | Define the address as `DestinationIPAddress`. |
| `Listener Mask` parameter | Define the mask as `DestinationNetmask` with a value of `255.255.255.255`. |
| `Load Balancing Method` parameter | Define the load balancing method by defining `LBMethod` with a value, such as "ROUND_ROBIN". |
| `InternalSelfIP > Enable Floating?` parameter | Set the value to `YES` or `NO`. Do not use floating unless you have an HA pair. |
| `InternalSelfIP > Port Lockdown` parameter | Set the value to `DEFAULT`. This parameter is mandatory. |
| `InternalSelfIP > Self IP Netmask` parameter | Set the value to `255.255.255.0`. |
| `ExternalSelfIP > Enable Floating?` parameter | Set the value to `YES` or `NO`. Do not use floating unless you have an HA pair. |
| `ExternalSelfIP > Port Lockdown` parameter | Set the value to `DEFAULT`. This parameter is mandatory. |
| `ExternalSelfIP > Self IP Netmask` parameter | Set the value to `255.255.255.0`. |
| `Pool Members` parameter | Associates the pool to the listener. |
| `EPGDestinationPort` parameter | Must be configured even though it has a mandatory value of `no`. |
| `EPGRatio` parameter | Must be configured even though it has a mandatory value of `no`. |
| `EPGConnectionLimit` parameter | Must be configured even though it has a mandatory value of `no`. |

| L4-L7 Parameter or Folder | Usage and Notes |
|---|---|
| `EPGConnectionRateLimit` parameter | Must be configured even though it has a mandatory value of `no`. |
| `LocalTraffic` folder | Serverfarm (pool) addresses of real servers (members). Change the name to something without "-Default". |
| `Network` folder | IP addresses of the F5 interfaces and the default route. In this folder, do the following things:<br>**1** Change the name to something without "-Default".<br>**2** Set the IP addresses of the external and internal interfaces.<br>**3** Add a static route on F5 that points to the VRF subnet. |
| `Listener` folder | The virtual server configuration. Make sure that the listener in on the same subnet as the bridge domain that connects the load balancer to the VRF.<br>In this folder, do the following things:<br>**1** Change the name to something without "-Default".<br>**2** Set the Protocol parameter.<br>**3** Set the Virtual Server IP Address parameter.<br>**4** Set the Virtual Server Netmask parameter.<br>**5** Set the Virtual Server Port parameter. |
| `Network Relationship` folder | In this folder, do the following things:<br>**1** Change the name to something without "-Default".<br>**2** Set the Select Network parameter and change the name to something without "-Default". This parameter points to the network configuration. |

| L4-L7 Parameter or Folder | Usage and Notes |
|---|---|
| `Pool` folder | Pool to be used by the virtual IP. In this folder, do the following things:<br>**1** Change the name to something without "-Default".<br>**2** Set the EPG Connect Rate Limit parameter.<br>**3** Set the EPG Connection Limit parameter.<br>**4** Set the EPG Destination Port parameter.<br>**5** Set the EPG Ratio parameter.<br>**6** Set the Select Pool parameter. This parameter points to the LocalTraffic configuration.<br>**7** Set the Pool Type parameter. Use **Dynamic** for dynamic endpoint attach, or **Static** for a predefined list of pool members.<br>**8** Define the pool members. The following parameters are necessary for the members:<br>• Connect Rate Limit<br>• Connection Limit<br>• Load Balancing Ratio<br>• Member IP Port<br>• MemberIP Address<br>**9** Set the Pool Monitor parameter.<br>**10** Set the Load Balancing Method parameter. |
| `Monitor` parameter | The Monitor parameter must be referenced from the pool. The Monitor configuration is mandatory, otherwise the Pool will not go up. |

The following XML is an example of a Layer 4 to Layer 7 parameters configuration:

```
<!-- Note: some parameters are mandatory: -->
<!-- such as the Listener, the Pool, the Monitor, the virtual address, the load balancing
method -->
<!-- without them the graph is not deployed -->

<!-- NETWORK FOLDER (called here Network1): SELFIP, STATIC ROUTE, NAT POOL -->
<!-- IP addresses for the F5 interfaces and default route we are just using the external
interface here -->
<!-- The IP of the external inteface must be on the same subnet as the BD that connects to
 the VRF -->
<vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="F5-GoTo" key="Network" locked="no"
 name="Network1" nodeNameOrLbl="F5-1-node" scopedBy="epg">
    <vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="F5-GoTo" key="ExternalSelfIP"
 locked="no" name="ExternalSelfIP" nodeNameOrLbl="F5-1-node">
        <vnsParamInst key="Floating" locked="no" name="Floating" value="NO"/>
```

```
        <vnsParamInst key="SelfIPNetmask" locked="no" name="SelfIPNetmask"
value="255.255.255.0"/>
       <vnsParamInst key="SelfIPAddress" locked="no" name="SelfIPAddress" value="10.0.0.3"/>

        <vnsParamInst key="PortLockdown" locked="no" name="PortLockdown" value="DEFAULT"/>
    </vnsFolderInst>

    <vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="F5-GoTo" key="InternalSelfIP"
 locked="no" name="InternalSelfIP" nodeNameOrLbl="F5-1-node">
        <vnsParamInst key="Floating" locked="no" name="Floating" value="NO"/>
        <vnsParamInst key="SelfIPNetmask" locked="no" name="SelfIPNetmask"
value="255.255.255.0"/>
       <vnsParamInst key="SelfIPAddress" locked="no" name="SelfIPAddress" value="30.0.0.3"/>

        <vnsParamInst key="PortLockdown" locked="no" name="PortLockdown" value="DEFAULT"/>
    </vnsFolderInst>

    <!-- STATIC ROUTE -->
   <vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="F5-GoTo" key="Route" locked="no"
 name="Route" nodeNameOrLbl="F5-1-node">
        <vnsParamInst key="DestinationIPAddress" locked="no" name="DestinationIPAddress"
value="0.0.0.0"/>
        <vnsParamInst key="DestinationNetmask" locked="no" name="DestinationNetmask"
value="0.0.0.0"/>
        <vnsParamInst key="NextHopIPAddress" locked="no" name="NextHopIPAddress"
value="10.0.0.2"/>
    </vnsFolderInst>

</vnsFolderInst>
<!-- END OF NETWORK FOLDER CONFIGURATION -->

<!-- LOCAL TRAFFIC FOLDER, called here LocalTrafficSSH -->
<!-- Definition of the load balancing mechanism, serverfarm and of monitoring-->
<vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="F5-GoTo" key="LocalTraffic"
locked="no" name="LocalTrafficSSH" nodeNameOrLbl="F5-1-node">

    <!-- CONFIGURE HERE SERVER MONITORING called here "ICMPMonitor"-->
    <vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="F5-GoTo" key="Monitor"
locked="no" name="ICMPMonitor" nodeNameOrLbl="F5-1-node">
        <vnsParamInst key="Type" locked="no" name="ICMP" value="ICMP"/>
        <vnsParamInst key="FailByAttempts" locked="no" name="FailByAttempts" value="3"/>
       <vnsParamInst key="FrequencySeconds" locked="no" name="FrequencySeconds" value="5"/>

    </vnsFolderInst>

    <!-- CONFIGURE HERE THE LIST OF SERVERS called here "ServerPoolSSH"-->
   <vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="F5-GoTo" key="Pool" locked="no"
 name="ServerPoolSSH" nodeNameOrLbl="F5-1-node">
        <!-- CONFIGURE HERE LOAD BALANCING TYPE -->
        <vnsParamInst key="LBMethod" locked="no" name="LBMethod" value="ROUND_ROBIN"/>
        <!-- Use Dynamic only if you want to use the EPG endpoints to autopopulate the
serverfarm pool -->
        <vnsParamInst key="PoolType" name="PoolType" value="DYNAMIC"/>

        <!-- Uncomment this section if you want to use statically defined pool members -->
```

```
        <!-- vnsParamInst key="PoolType" locked="no" name="PoolType" value="STATIC"/>
        <!-- First Server in the Pool: Member1 -->
        <!-- vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="F5-GoTo" key="Member"
 locked="no" name="Member1" nodeNameOrLbl="F5-1-node"-->
            <!-- vnsParamInst key="Port" name="Port" value="22"/-->
            <!--vnsParamInst key="IPAddress" name="IPAddress" value="30.0.0.101"/-->
        <!-- /vnsFolderInst -->

        <!-- Second Server in the Pool: Member2 -->
        <!--vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="F5-GoTo" key="Member"
locked="no" name="Member2"  nodeNameOrLbl="F5-1-node"-->
            <!--vnsParamInst key="Port" name="Port" value="22"/-->
            <!--vnsParamInst key="IPAddress" name="IPAddress" value="30.0.0.102"/-->
        <!--/vnsFolderInst>


        <!-- This is a relation to the Pool Monitoring defined in "Monitor" -->
       <vnsFolderInst ctrctNameOrLbl="webtoapp"  graphNameOrLbl="F5-GoTo" key="PoolMonitor"
  locked="no" name="ArbitraryNamePoolMonitor" nodeNameOrLbl="F5-1-node">
            <vnsCfgRelInst key="PoolMonitorRel" locked="no" name="PoolMonitorRel"
targetName="LocalTrafficSSH/ICMPMonitor"/>
        </vnsFolderInst>
    </vnsFolderInst>

</vnsFolderInst>
<!-- END OF LOCAL TRAFFIC FOLDER -->

<!-- MAIN FUNCTION CONFIG: here you define the Virtual IP, The Serverfarm, which Network
config you want to use -->
<!-- Virtual IP for F5 (LISTENER) -->
<vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="F5-GoTo" key="Listener" locked="no"
 name="ListenerSSH" nodeNameOrLbl="F5-1-node">
    <vnsParamInst key="DestinationPort" locked="no" name="DestinationPort" value="22"/>
    <vnsParamInst key="Protocol" locked="no" name="Protocol" value="TCP"/>
    <vnsParamInst key="DestinationNetmask" locked="no" name="DestinationNetmask"
value="255.255.255.255"/>
    <vnsParamInst key="DestinationIPAddress" locked="no" name="DestinationIPAddress"
value="10.0.0.80"/>

</vnsFolderInst>

<!-- Relation to the Serverfarm for F5 (which is defined within the LOCAL TRAFFIC)-->
<!-- This has a relation to "ServerPoolSSH" -->
<!-- If you don't put this configuration the Listener doesn't have any severfarm pool
associated with it -->
<vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="F5-GoTo" key="Pool" locked="no"
name="ArbitraryNameServerFarm" nodeNameOrLbl="F5-1-node">
      <vnsCfgRelInst key="PoolRel" locked="no" name="SSHserversforListenerSSH"
targetName="LocalTrafficSSH/ServerPoolSSH"/>

      <!-- The following parameters are necessary -->
      <!-- PLS CHANGE THE L4 DESTINATION PORT AS NECESSARY -->
       <vnsParamInst name="EPGDestinationPort"  key="EPGDestinationPort" value="22"
mandatory="no" />
       <vnsParamInst name="EPGRatio" locked="no" key="EPGRatio" value="1"  mandatory="no"
```

```
/>
       <vnsParamInst name="EPGConnectionLimit"  key="EPGConnectionLimit"
cardinality="unspecified" value="1000" mandatory="no" />
      <vnsParamInst name="EPGConnectionRateLimit" key="EPGConnectionRateLimit" value="1000"
 mandatory="no" />
</vnsFolderInst>

<!-- Network Relation for F5 -->
<!-- This defines the network configuration for this virtual server instance -->
<vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="F5-GoTo" key="NetworkRelation"
locked="no" name="ArbitraryNameNetworkRelation" nodeNameOrLbl="F5-1-node">
    <vnsCfgRelInst key="NetworkRel" locked="no" name="NetworkRel" targetName="Network1"/>
</vnsFolderInst>
```

**Step 6** Create a service graph template and either use a function profile or enter the Layer 4 to Layer 7 parameters by hand.
See Creating a Layer 4 to Layer 7 Service Graph Template Using the GUI, on page 44.

a) Drag the defined logical device to the canvas.
b) In the **F5Cluster Information** section, for the **ADC** radio buttons, choose **Two-Arm**.

**Step 7** Apply the service graph template.
See Applying a Service Graph Template to Endpoint Groups Using the GUI, on page 45.

You cannot configure an "any" virtual IP or port. You can only choose **TCP** or **UDP** option; there is no "all IP protocol" value.

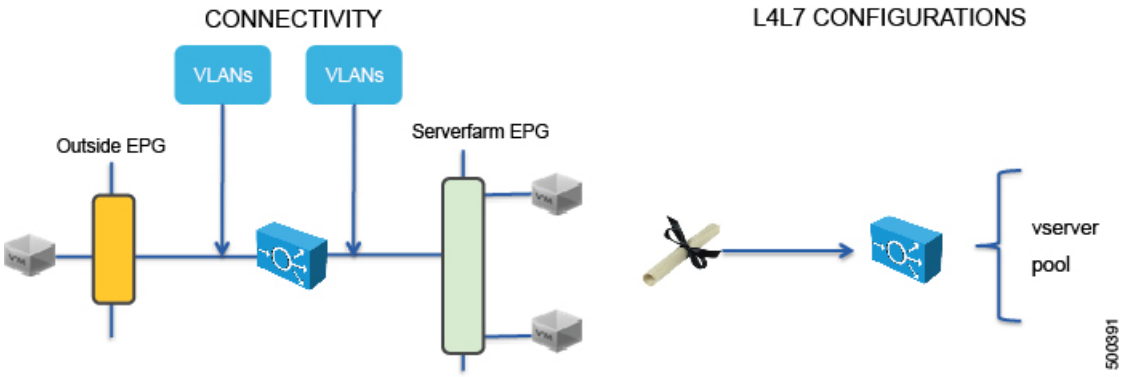
**Step 8** Verify that the configuration deployed successfully.
See Verifying the Configuration for an F5 Device, on page 78.

# F5 in One-Arm Mode

## About Deploying F5 in One-Arm Mode

This section explains how to deploy an F5 device in one-arm mode as part of the service graph. As in all service graphs, the service graph with F5 in one-arm mode is still defined as a contract connecting two endpoint groups (EPGs)—the outside EPG and and the serverfarm EPG, as illustrated in the following figure:

***Figure 49: Service Graph with F5 in One-Arm Mode***

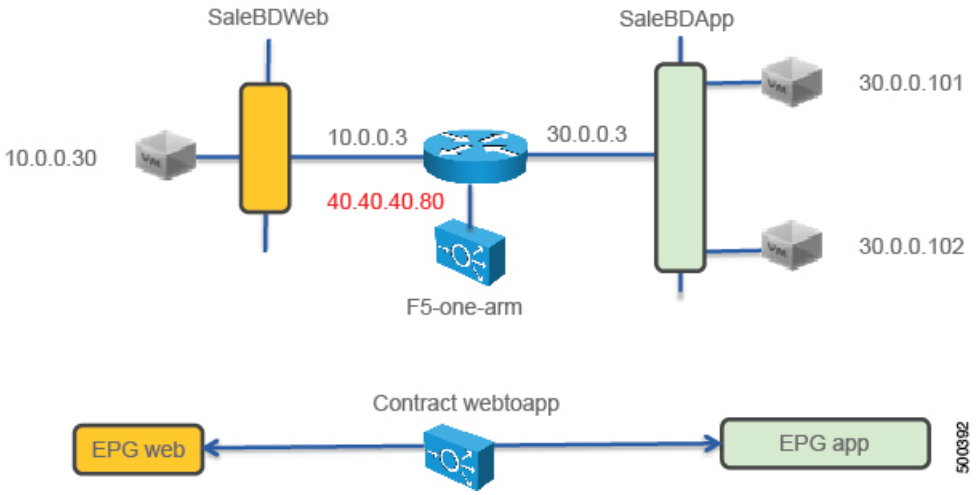



The service graph also defines the Layer 4 to Layer 7 configurations that must be loaded onto the Layer 4 to Layer 7 device.

When deploying the F5 load balancer in one-arm mode, the contract is still defined between two endpoint groups, such as web and app as in Figure 50: Logical Topology for an F5 One-Arm Deployment, on page 69. The endpoint groups are associated with two bridge domains, such as SaleBDWeb (10.0.0.0/24) and

SaleBDApp (30.0.0.0/24). The main difference with the other service graph modes is that the F5 device is attached to another bridge domain (40.40.40.0/24).

***Figure 50: Logical Topology for an F5 One-Arm Deployment***



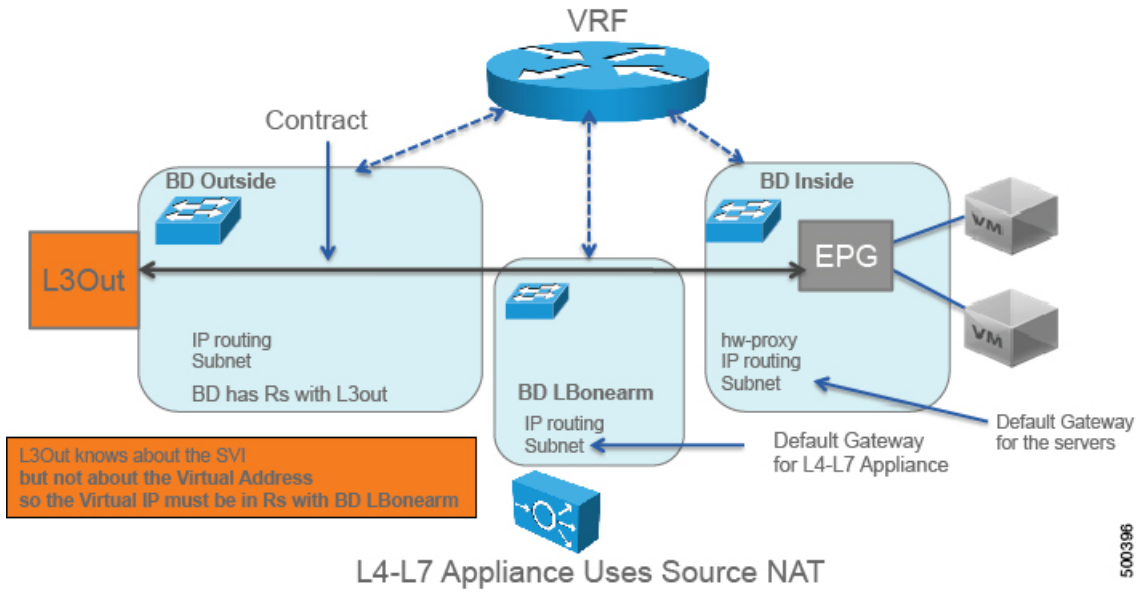

To deploy an F5 device in the one-arm mode, you must do the following things:

- Configure 3 bridge domains with unicast routing enabled
- Configure the F5 device as a GoTo device
- Configure source NAT on the F5 device
- Configure a VIP on the same subnet as the bridge domain to which the F5 connects
- Configure the contract between the outside and inside endpoint group
- Associate the service graph with the contract
- Configure the logical interfaces external and internal to point to the same "internal" interface (1_2)

  You could instead use the "external" interface. However, this document indicates to use the "internal" interface because if you want to enable endpoint attach, you must associate the internal interface to the bridge domain LBonearm.
- Configure the logical device context to use the same bridge domain as the one connecting F5 to the VRF

The following figure illustrates the topology of the 3 bridge domains:

***Figure 51: The Three Bridge Domains for an F5 One-Arm Deployment***




# Overview of Preparing an F5 Device in One-Arm Mode

The following procedure provides of overview of preparing an F5 device to be deployed in one-arm mode.

### Procedure

**Step 1** In the Application Policy Infrastructure Controller (APIC), define the VLAN pool to use.

**Step 2** Create a virtual domain.

**Step 3** Create the attach entity profile.

**Step 4** Download the device package from the F5 Web site.

**Step 5** Upload the device package to the APIC.

# Deploying F5 in One-Arm Mode

The tasks that you must perform to deploy F5 in one-arm mode are nearly identical to the tasks for generically deploying a service graph, with a few differences. The following procedure provides the generic service graph deployment tasks, along with information about what you must do differently to deploy F5 in one-arm mode.

### Procedure

**Step 1** Create physical and virtual domains.

**Step 2** Configure the basic management access on the Layer 4 to Layer 7 device.

**Step 3** Import the device package.
See Importing a Device Package Using the GUI, on page 35.

**Step 4** Create the bridge domains and VRFs.
See Creating Bridge Domains and VRFs Using the GUI, on page 36.

You must create 3 bridge domains:

- One for the client side (external).

  On this bridge domain:

  - Enable unicast routing.

- One for the server side. The subnet on this bridge domain is the default gateway for the servers.

  On this bridge domain:

  - Enable unicast routing.

- One for the load balancer, only. The service graph template wizard will create the association with it. The subnet on this bridge domain is the default gateway for the load balancer.

  On this bridge domain:

  - Enable ARP flooding.
  - Enable unicast routing.
  - Set the MAC unknown unicast destination action to "flood".
  - Set the subnet IP address and configure it as the default gateway on the F5 device.

**Step 5** Create endpoint groups and contracts.
See Creating Endpoint Groups and Contracts Using the GUI, on page 37.

**Step 6** Configure logical devices and concrete devices.
See Creating a Logical or Concrete Device Using the GUI, on page 39.

a) For a logical device you must define interface `1_2` for both external and internal.
The following XML is an example of defining interface `1_2`:

```
<vnsLIf name="F5ClusterExt">
...
    <vnsRsMetaIf tDn="uni/infra/mDev-F5-BIGIP-2.0/mIfLbl-external"/>
    <vnsRsCIfAtt tDn="uni/tn-Sales/lDevVip-F5Cluster/cDev-BIGIP-1/cIf-[1_2]"/>
</vnsLIf>
<vnsLIf name="F5ClusterInt">
    <vnsRsMetaIf tDn="uni/infra/mDev-F5-BIGIP-2.0/mIfLbl-internal"/>
    <vnsRsCIfAtt tDn="uni/tn-Sales/lDevVip-F5Cluster/cDev-BIGIP-1/cIf-[1_2]"/>
</vnsLIf>
```

b) For the concrete device, in the **Service Type** drop-down list, choose **ADC** for a load balancer.
c) If the device is virtual, in the **VMM Domain** drop-down list, choose the appropriate VMM domain.
d) In the **Model** drop-down list, choose **BIG-IP-VE-GENERIC** for an F5 VE.
e) In the **Function Type** buttons, click **GoTo**.
f) In the **Cluster** section, for **Cluster Interfaces**, add the same interface twice. For the **Type**, choose **consumer** for one interface and **provider** for the other interface.
g) For the Layer 4 to Layer 7 parameters, for the Host Name parameter, set the value to the host name of the F5 device.

**Step 7** Create or import a function profile.
See Creating a Function Profile Using the GUI, on page 43 or Importing a Function Profile Using the GUI, on page 44.

- The Layer 4 to Layer 7 parameters under the `CDev` object refer to the common partition of F5; you do not need to use these parameters
- All the parameters with "-Default" in the value must be changed to something else, such as "Pool" or "Listener"
- If you use HA, you cannot use the GUI to configure the parameters because the SelfIP parameter value will be the same on the active and on the standby appliance

The following table describes the mandatory Layer 4 to Layer 7 parameters and provides examples of possible values that you must change for your specific configuration:

***Table 2: Layer 4 to Layer 7 Parameters for F5 in One-Arm Mode***

| L4-L7 Parameter or Folder | Usage and Notes |
|---|---|
| `Listener IP` folder | Define the address as `DestinationIPAddress`. |
| `Listener Mask` parameter | Define the mask as `DestinationNetmask` with a value of `255.255.255.255`. |
| `Load Balancing Method` parameter | Define the load balancing method by defining `LBMethod` with a value, such as "ROUND_ROBIN". |
| `InternalSelfIP > Enable Floating?` parameter | Set the value to `YES` or `NO`. Do not use floating unless you have an HA pair.<br>You do not need to configure both InternalSelfIP and ExternalSelfIP. |
| `InternalSelfIP > Self IP Netmask` parameter | Set the value to `255.255.255.0`. |
| `Pool Members` parameter | Associates the pool to the listener. |
| `EPGDestinationPort` parameter | Must be configured even though it has a mandatory value of `no`. |
| `EPGRatio` parameter | Must be configured even though it has a mandatory value of `no`. |

| L4-L7 Parameter or Folder | Usage and Notes |
|---|---|
| `EPGConnectionLimit` parameter | Must be configured even though it has a mandatory value of `no`. |
| `EPGConnectionRateLimit` parameter | Must be configured even though it has a mandatory value of `no`. |
| `LocalTraffic` folder | Serverfarm (pool) addresses of real servers (members). Change the name to something without "-Default". |
| `Network` folder | IP addresses of the F5 interfaces and the default route. In this folder, do the following things:<br>**1** Change the name to something without "-Default".<br>**2** Set the IP addresses of the external and internal interfaces.<br>**3** Add a static route on F5 that points to the VRF subnet. |
| `Listener` folder | The virtual server configuration. Make sure that the listener in on the same subnet as the bridge domain that connects the load balancer to the VRF.<br>In this folder, do the following things:<br>**1** Change the name to something without "-Default".<br>**2** Set the Protocol parameter.<br>**3** Set the Virtual Server IP Address parameter.<br>**4** Set the Virtual Server Netmask parameter.<br>**5** Set the Virtual Server Port parameter.<br>**6** Set the SNAT > SNAT Type parameter to `automap`. |
| `Network Relationship` folder | In this folder, do the following things:<br>**1** Change the name to something without "-Default".<br>**2** Set the Select Network parameter and change the name to something without "-Default". This parameter points to the network configuration. |

| L4-L7 Parameter or Folder | Usage and Notes |
|---|---|
| `Pool` folder | Pool to be used by the virtual IP. In this folder, do the following things:<br>**1** Change the name to something without "-Default".<br>**2** Set the EPG Connect Rate Limit parameter.<br>**3** Set the EPG Connection Limit parameter.<br>**4** Set the EPG Destination Port parameter.<br>**5** Set the EPG Ratio parameter.<br>**6** Set the Select Pool parameter. This parameter points to the LocalTraffic configuration.<br>**7** Set the Pool Type parameter. Use **Dynamic** for dynamic endpoint attach, or **Static** for a predefined list of pool members.<br>**8** Define the pool members. The following parameters are necessary for the members:<br>• Connect Rate Limit<br>• Connection Limit<br>• Load Balancing Ratio<br>• Member IP Port<br>• MemberIP Address<br>**9** Set the Pool Monitor parameter.<br>**10** Set the Load Balancing Method parameter. |
| `Monitor` parameter | The Monitor parameter must be referenced from the pool. The Monitor configuration is mandatory, otherwise the Pool will not go up. |

The following XML is an example of a Layer 4 to Layer 7 parameters configuration:

```
<!-- Note: some parameters are mandatory: -->
<!-- such as the Listener, the Pool, the Monitor, the virtual address, the load balancing
method -->
<!-- without them the graph is not deployed -->

<!-- NETWORK FOLDER (called here Network1): SELFIP, STATIC ROUTE, NAT POOL -->
<!-- IP addresses for the F5 interfaces and default route we are just using the external
interface here -->
<!-- The IP of the external inteface must be on the same subnet as the BD that connects to
 the VRF -->
<!-- In one-arm mode the configuration must use InternalSelfIP -->
<vnsFolderInst  ctrctNameOrLbl="webtoapp" graphNameOrLbl="F5-onearm" key="Network" locked="no"
 name="Network1" nodeNameOrLbl="F5-1-node" scopedBy="epg">
    <vnsFolderInst  ctrctNameOrLbl="webtoapp" graphNameOrLbl="F5-onearm" key="InternalSelfIP"
 locked="no" name="InternalSelfIP" nodeNameOrLbl="F5-1-node" scopedBy="epg">
```

```
        <vnsParamInst  key="Floating" locked="no"  name="Floating"  value="NO"/>
        <vnsParamInst  key="SelfIPNetmask" locked="no"  name="SelfIPNetmask"
value="255.255.255.0"/>
        <vnsParamInst  key="SelfIPAddress" locked="no"  name="SelfIPAddress"
value="40.40.40.5"/>
       <vnsParamInst  key="PortLockdown" locked="no"  name="PortLockdown"  value="DEFAULT"/>

    </vnsFolderInst>

    <!-- STATIC ROUTE -->
    <vnsFolderInst  ctrctNameOrLbl="webtoapp" graphNameOrLbl="F5-onearm" key="Route"
locked="no" name="Route" nodeNameOrLbl="F5-1-node" scopedBy="epg">
        <vnsParamInst  key="DestinationIPAddress" locked="no"  name="DestinationIPAddress"
  value="0.0.0.0"/>
        <vnsParamInst  key="DestinationNetmask" locked="no"  name="DestinationNetmask"
value="0.0.0.0"/>
        <vnsParamInst  key="NextHopIPAddress" locked="no"  name="NextHopIPAddress"
value="40.40.40.3"/>
    </vnsFolderInst>

    <!-- SNAT IP ADDRESS, SNATPool1 is an arbitrary name referenced later -->
    <vnsFolderInst  ctrctNameOrLbl="webtoapp" graphNameOrLbl="F5-onearm" key="SNATPool"
locked="no" name="SNATPool1" nodeNameOrLbl="F5-1-node" scopedBy="epg">
        <vnsParamInst   key="SNATIPAddress"  name="SNATIPAddress" value="40.40.40.10"/>
    </vnsFolderInst>
</vnsFolderInst>
<!-- END OF NETWORK FOLDER CONFIGURATION -->

<!-- LOCAL TRAFFIC FOLDER, called here LocalTrafficSSH -->
<!-- Definition of the load balancing mechanism, serverfarm and of monitoring-->
<vnsFolderInst  ctrctNameOrLbl="webtoapp" graphNameOrLbl="F5-onearm" key="LocalTraffic"
locked="no" name="LocalTrafficSSH" nodeNameOrLbl="F5-1-node" scopedBy="epg">

    <!-- CONFIGURE HERE SERVER MONITORING called here "ICMPMonitor"-->
    <vnsFolderInst  ctrctNameOrLbl="webtoapp" graphNameOrLbl="F5-onearm" key="Monitor"
locked="no" name="ICMPMonitor" nodeNameOrLbl="F5-1-node" scopedBy="epg">
        <vnsParamInst  key="Type" locked="no"  name="ICMP"  value="ICMP"/>
       <vnsParamInst  key="FailByAttempts" locked="no"  name="FailByAttempts"  value="3"/>

        <vnsParamInst  key="FrequencySeconds" locked="no"  name="FrequencySeconds"
value="5"/>
    </vnsFolderInst>

    <!-- CONFIGURE HERE THE LIST OF SERVERS called here "ServerPoolSSH"-->
    <vnsFolderInst  ctrctNameOrLbl="webtoapp" graphNameOrLbl="F5-onearm" key="Pool"
locked="no" name="ServerPoolSSH" nodeNameOrLbl="F5-1-node" scopedBy="epg">
        <!-- CONFIGURE HERE LOAD BALANCING TYPE -->
        <vnsParamInst  key="LBMethod" locked="no"  name="LBMethod"  value="ROUND_ROBIN"/>
        <!-- Use Dynamic only if you want to use the EPG endpoints to autopopulate the
serverfarm pool -->
        <vnsParamInst  key="PoolType" locked="no"  name="PoolType"  value="STATIC"/>

        <!-- First Server in the Pool: Member1 -->
        <vnsFolderInst  ctrctNameOrLbl="webtoapp" graphNameOrLbl="F5-onearm" key="Member"
locked="no" name="Member1" nodeNameOrLbl="F5-1-node">
```

```
            <vnsParamInst  key="Port" name="Port" value="22"/>
            <vnsParamInst  key="IPAddress" name="IPAddress" value="30.0.0.101"/>
        </vnsFolderInst>

        <!-- Second Server in the Pool: Member2 -->
        <vnsFolderInst  ctrctNameOrLbl="webtoapp" graphNameOrLbl="F5-onearm" key="Member"
locked="no" name="Member2"  nodeNameOrLbl="F5-1-node">
            <vnsParamInst   key="Port" name="Port" value="22"/>
            <vnsParamInst   key="IPAddress" name="IPAddress" value="30.0.0.102"/>
        </vnsFolderInst>

        <!-- This is a relation to the Pool Monitoring defined in "Monitor" -->
        <vnsFolderInst  ctrctNameOrLbl="webtoapp"  graphNameOrLbl="F5-onearm"
key="PoolMonitor"  locked="no"  name="ArbitraryNamePoolMonitor" nodeNameOrLbl="F5-1-node">
            <vnsCfgRelInst  key="PoolMonitorRel" locked="no" name="PoolMonitorRel"
targetName="LocalTrafficSSH/ICMPMonitor"/>
        </vnsFolderInst>
    </vnsFolderInst>

</vnsFolderInst>
<!-- END OF LOCAL TRAFFIC FOLDER -->

<!-- MAIN FUNCTION CONFIG: here you define the Virtual IP, The Serverfarm, which Network
config you want to use and the SNAT -->
<!-- Virtual IP for F5 (LISTENER) -->
<!-- In one-arm mode this must be on the same subnet as the BD that connects to the VRF -->
<vnsFolderInst  ctrctNameOrLbl="webtoapp" graphNameOrLbl="F5-onearm" key="Listener"
locked="no" name="ListenerSSH" nodeNameOrLbl="F5-1-node" scopedBy="epg">
    <vnsParamInst  key="DestinationPort" locked="no"  name="DestinationPort" value="22"/>
    <vnsParamInst  key="Protocol" locked="no"  name="Protocol"  value="TCP"/>
    <vnsParamInst  key="DestinationNetmask" locked="no"  name="DestinationNetmask"
value="255.255.255.255"/>
    <vnsParamInst  key="DestinationIPAddress" locked="no"  name="DestinationIPAddress"
value="40.40.40.80"/>

    <!-- This Virtual IP uses the source NAT from Network1 called SNATPool1-->
   <vnsFolderInst  ctrctNameOrLbl="webtoapp"   graphNameOrLbl="F5-onearm" key="SNAT"
name="ArbitraryNameNAT" nodeNameOrLbl="F5-1-node">
      <vnsCfgRelInst key="SNATRel" name="SNATRel" targetName="Network1/SNATPool1"/>
      <vnsParamInst  key="SNATType" name="SNATType" value="SNAT_POOL"/>
   </vnsFolderInst>

</vnsFolderInst>

<!-- Relation to the Serverfarm for F5 (which is defined within the LOCAL TRAFFIC)-->
<!-- This has a relation to "ServerPoolSSH" -->
<!-- If you don't put this configuration the Listener doesn't have any severfarm pool
associated with it -->
<vnsFolderInst  ctrctNameOrLbl="webtoapp" graphNameOrLbl="F5-onearm" key="Pool" locked="no"
 name="ArbitraryNameServerFarm" nodeNameOrLbl="F5-1-node" scopedBy="epg">
   <vnsCfgRelInst  key="PoolRel" locked="no" name="SSHserversforListenerSSH"
targetName="LocalTrafficSSH/ServerPoolSSH"/>
    <!-- These parameters are necessary -->
   <!-- Change the destination L4PORT as needed  -->
    <vnsParamInst name="EPGDestinationPort" key="EPGDestinationPort" value="22" mandatory="no"
```

```
 />
    <vnsParamInst name="EPGRatio" locked="no" key="EPGRatio" cardinality="unspecified"
value="1" mandatory="no" />
    <vnsParamInst name="EPGConnectionLimit" key="EPGConnectionLimit" value="1000"
mandatory="no" />
    <vnsParamInst name="EPGConnectionRateLimit" key="EPGConnectionRateLimit" value="1000"
mandatory="no" />

</vnsFolderInst>

<!-- Network Relation for F5 -->
<!-- This defines the network configuration for this virtual server instance -->
<vnsFolderInst  ctrctNameOrLbl="webtoapp" graphNameOrLbl="F5-onearm" key="NetworkRelation"
 locked="no" name="ArbitraryNameNetworkRelation" nodeNameOrLbl="F5-1-node" scopedBy="epg">
   <vnsCfgRelInst  key="NetworkRel" locked="no"  name="NetworkRel" targetName="Network1"/>
</vnsFolderInst>


<!-- END OF MAIN FUNCTION CONFIG -->
```

**Step 8** Create a service graph template and either use a function profile or enter the Layer 4 to Layer 7 parameters by hand.
See Creating a Layer 4 to Layer 7 Service Graph Template Using the GUI, on page 44.

a) Drag the defined logical device to the canvas.
b) In the **F5Cluster Information** section, for the **ADC** radio buttons, choose **One-Arm**.

**Step 9** Apply the service graph template.
See Applying a Service Graph Template to Endpoint Groups Using the GUI, on page 45.

You cannot configure an "any" virtual IP or port. You can only choose **TCP** or **UDP** option; there is no "all IP protocol" value.

**Step 10** Create the logical device context (optional if you used the GUI wizard).
See Creating a Device Selection Policy Using the GUI, on page 50.

The following XML is an example of defining a logical device context:

```
<!-- Connector name is defined in the Abstract Graph -->
<vnsLIfCtx connNameOrLbl="F5nodeclientside">
    <vnsRsLIfCtxToLIf tDn="uni/tn-Sales/lDevVip-F5Cluster/lIf-F5ClusterExt"/>
    <vnsRsLIfCtxToBD tDn="uni/tn-Sales/BD-LBBD"/>
</vnsLIfCtx>

<!-- Connector name is defined in the Abstract Graph -->
<vnsLIfCtx connNameOrLbl="F5nodeserverside">
    <vnsRsLIfCtxToLIf tDn="uni/tn-Sales/lDevVip-F5Cluster/lIf-F5ClusterInt"/>
    <vnsRsLIfCtxToBD tDn="uni/tn-Sales/BD-LBBD"/>
</vnsLIfCtx>
```

The highlighted lines select the bridge domain that connects to the VRF.

**Step 11** Verify that the configuration deployed successfully.
See Verifying the Configuration for an F5 Device, on page 78.

# Verifying the Configuration for an F5 Device

After you deployed an F5 device in any mode, you can verify that the configuration is functioning properly by using the following procedure:

### Procedure

**Step 1** In the Application Policy Infrastructure Controller (APIC) GUI, on the menu bar, choose **Tenants** > **All Tenants**.

**Step 2** In the Work pane, double click the tenant's name.

**Step 3** In the Navigation pane, choose **Tenant** ***tenant_name*** > **L4-L7 Services** > **Deployed Graph Instances** > ***F5_graph_name***.

**Step 4** In the Work pane, examine the F5 service graph's state. If the F5 device displays a green light indicator, then the service graph was deployed successfully.

**Step 5** In the Navigation pane, choose **Tenant** ***tenant_name*** > **L4-L7 Services** > **Deployed Devices** > ***F5_device_name***.

**Step 6** In the Work pane, view the F5 device's properties.

- The health score should be 100.
- In the **Properties** section, the **Virtual Device ID** should match the partition number in F5.

**Step 7** In the Navigation pane, choose **Tenant** ***tenant_name*** > **L4-L7 Services** > **Deployed Devices** > ***F5_device_name*** > ***virtual_server_name***.
The virtual server is the listener.

**Step 8** In the Work pane, view the virtual server's properties.

- The health score should be 100.

**Step 9** In the Navigation pane, choose **Tenant** ***tenant_name*** > **Application Profiles** > ***application_profile_name*** > **Application EPGs** > ***EPG_name*****L4-L7 Service Parameters**.

**Step 10** Verify that the Layer 4 to Layer 7 service parameters are set properly.

# Undoing a Service Graph Configuration for F5

You can undo a service graph configuration for F5 by using the following procedure:

### Procedure

**Step 1** In the Application Policy Infrastructure Controller (APIC) GUI, delete the service graph template.
See Undoing a Service Graph Configuration Using the GUI, on page 49.

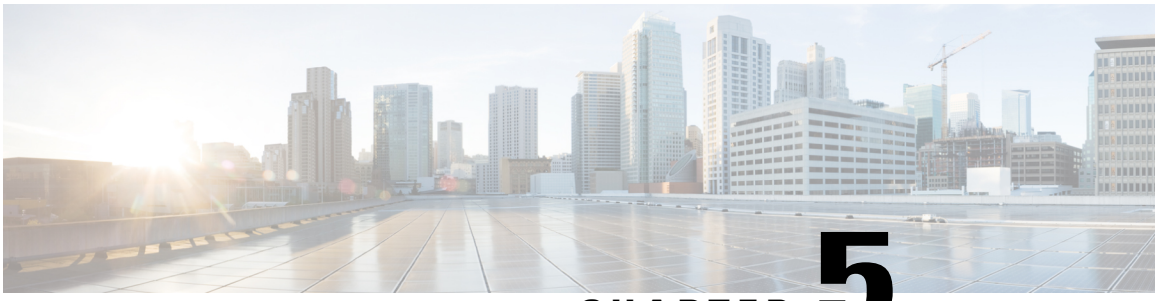
The F5 partitions are removed automatically, but you can remove them manually.

**Step 2** (Optional) To delete the partitions using the F5 GUI:

a) Change to the common partition.
b) Delete the partitions that you used for the F5 deployment.

**Step 3** (Optional) To delete the partitions using the F5 CLI:

a) Use **ssh** to log into the F5 device.
b) Change to the `/config/partitions` directory:
```
cd /config/partitions
```
c) Delete the partitions that you used for the F5 deployment:
```
rm -r apic_xxx*
```
d) Reload the system configuration in all partitions:
```
tmsh load sys-config partitions all
```



CHAPTER 5

# Deploying ASA



## ASA Deployment Modes in ACI Fabric

The following ASA deployment modes are supported in the Cisco Application Centric Infrastructure (ACI) fabric:

- Single context and multiple context modes with the ASA device package version 1.2 or later.

  Both context modes use VLAN sub-interfaces to separate traffic of different tenants.

- Transparent (bump in the wire) mode for "GoThrough" insertion.
  - Forwarding is done based on MAC address, but the routing table is needed for NAT and application inspection.
  - Flooding must be enabled in the ACI bridge domains.

- Routed (Layer 3 hop) mode for "GoTo" insertion.

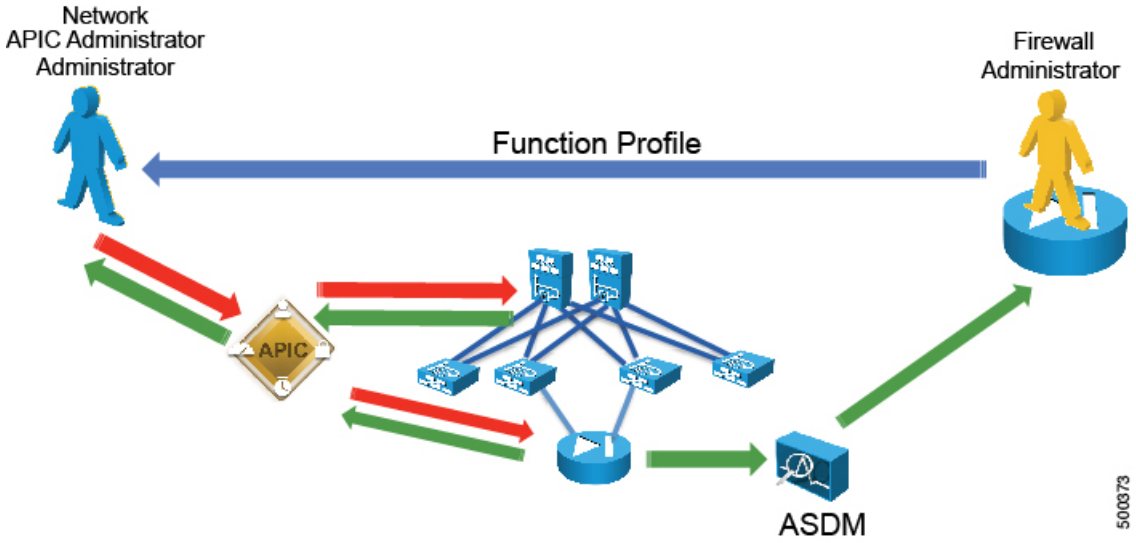
  You can configure only a single routing table per context, so you must specify destination static routes for target endpoint group subnets, or you can use dynamic routing with the ASA device package version 1.2 or later.

**Note** You do not need to create multiple contexts to create multiple service graphs. You can create multiple service graphs within a single context as long as the interface and ACL names are unique.

# About the ASA Operational Model

In the ASA operational model, the ASA configuration is managed through the Application Policy Infrastructure Controller (APIC). The following figure illustrates the ASA operational model:

***Figure 52: ASA Operational Model***




The ASA administrator provides the XML or JSON function profile configuration to the APIC administrator who then pushes the function profile through the APIC to the ASA device.

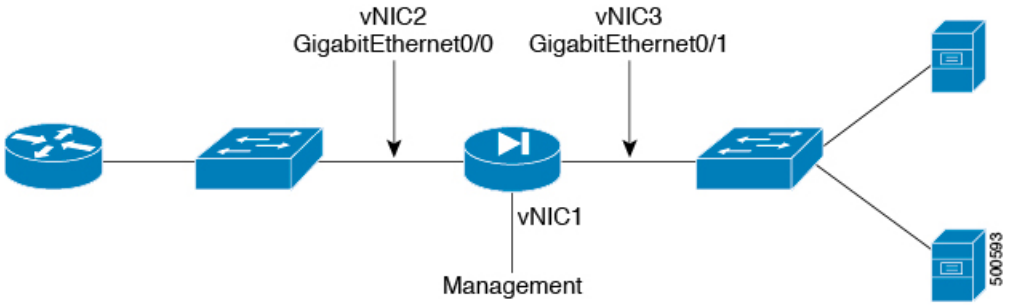
# Translation of ASA Terminology

The following table translates which vNIC corresponds to which interface in Cisco ASAv and Cisco Application Centric Infrastructure (ACI):

| Interface | VMware | ASAv | ACI | IP address is entered as |
|---|---|---|---|---|
| Management | vNIC1 | Management | N/A | N/A |
| Outside | vNIC2 | GigabitEthernet0/0 | GigabitEthernet0/0 | ExternalIf |
| Inside | vNIC3 | GigabitEthernet0/1 | GigabitEthernet0/1 | InternalIf |

The following figure illustrates the naming convention for the interfaces in the case of a Cisco ASAv firewall:

***Figure 53: ASAv Firewall Interface Naming Convention***




# About ASA Multi-Context Mode

You can partition a single physical ASA into multiple virtual firewalls, known as security/virtual contexts. Each context acts as an independent device with its own security policy, interfaces, and management IP address. You can use ASA multi-context capability in Cisco Application Centric Infrastructure (ACI) along with an ASA service graph. This configuration is supported with an ASA 5500-X device and ASA device package 1.2 or later.

ASA supports multi-context by adding individual ASA contexts as `Cdev` objects under the Layer 4 to Layer 7 devices. The Layer 4 to Layer 7 parameter configuration is pushed to individual ASA contexts, not to the "Admin" context. ACI pushes the "allocate-interface" configuration to the "Admin" context for the other contexts. This means that the IP address of the "Admin" context must be entered as the management IP address for the logical device configuration.

# About ASA High Availability and Scalability

Failover provides simple device-level redundancy. Failover peers are adjacent on data interfaces and have the same active IP address or MAC address. An active/standby failover pair can be initially configured and managed through the Application Policy Infrastructure Controller (APIC). You must first register both ASAs with the APIC. Active/active failover not supported.

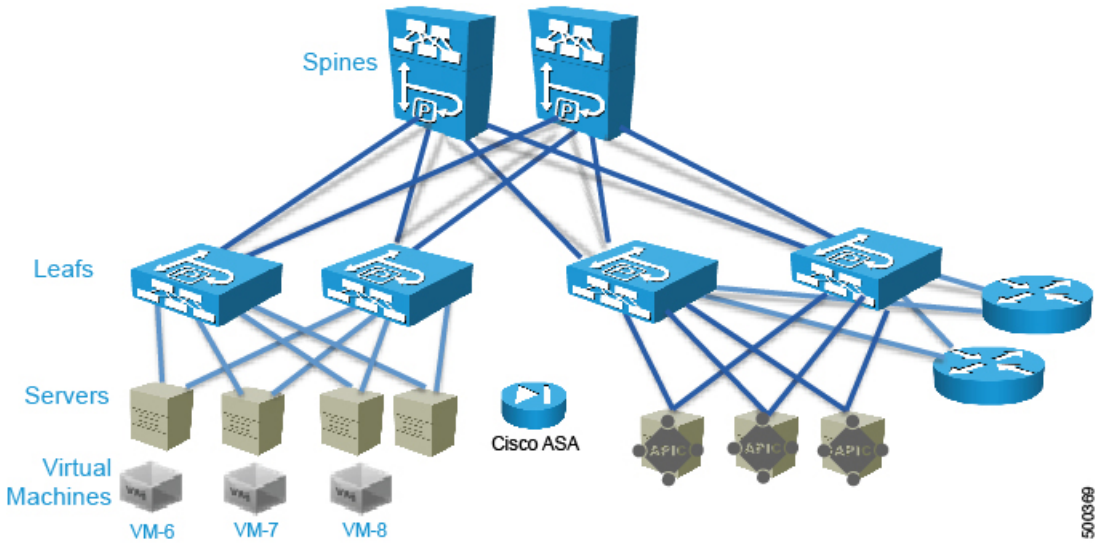
The ASA cluster must be deployed out of band, but the APIC can manage the cluster once it is deployed.

# ASA in GoTo Mode

## About Deploying ASA in GoTo Mode

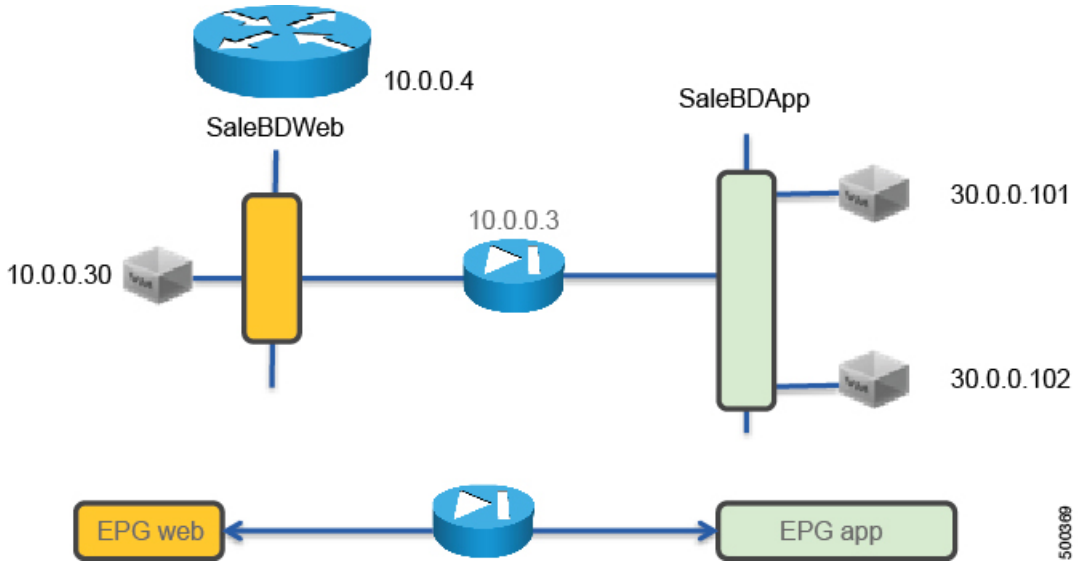
The following figure illustrates the topology for deploying Cisco Application Centric Infrastructure (ACI) fabric with ASA devices:

***Figure 54: ACI Fabric with ASA Devices***




The following figure illustrates the logical topology of an ASA GoTo deployment:

***Figure 55: Logical Topology of an ASA GoTo Deployment***



To deploy an ASA device in the GoTo mode, you must do the following things:

- Configure 2 bridge domains
- Configure 2 endpoint groups with each one associated with a different bridge domain
- Configure the ASA device as a GoTo device
- Set up NAT with a public IP on the same subnet as the bridge domain that ASA connects to on the outside (or consumer side)
- Configure the contract between the outside and inside endpoint group (or server side or provider side)
- Associate the service graph with the contract
- Associate the external logical interface with GigabitEthernet0/0 (which in the case of ASAv is Network Adapter 2)
- Associate the internal logical interface with GigabitEthernet0/1 (which in the case of ASAv is Network Adapter 3)

# Overview of Preparing an ASA Device in GoTo Mode

ASA and ASAv do not have the concept of VRF management. If ASA or ASAv are deployed in GoTo mode you might want to use "inband" management to ASA to avoid conflicting entries in the routing table. If you are using the service device in transparent mode, you do not need to use "inband" management because there is no need for VRF management.

The following procedure provides of overview of preparing an ASA device to be deployed in GoTo mode.

### Procedure

**Step 1** Enable SSH.

**Step 2** Enable HTTP access.

**Step 3** Configure the credentials.
You do not need to configure the interfaces, VLANs, or IP addresses.

**Step 4** Enter the following commands to create the initial configuration:

```
asa1(config)# no firewall transparent
asa1(config)# Interface Management0/0
asa1(config)# nameif management
asa1(config)# no shut
asa1(config)# hostname ASAv
asa1(config)# route management 0.0.0.0 0.0.0.0 192.168.11.254
asa1(config)# user admin password tme12345
asa1(config)# enable password tme12345
asa1(config)# aaa authentication ssh console LOCAL
asa1(config)# http server enable
asa1(config)# http 0.0.0.0 0.0.0.0 management
asa1(config)# ssh 0.0.0.0 0.0.0.0 management
```

# Configuring Bridge Domains for ASA in GoTo Mode

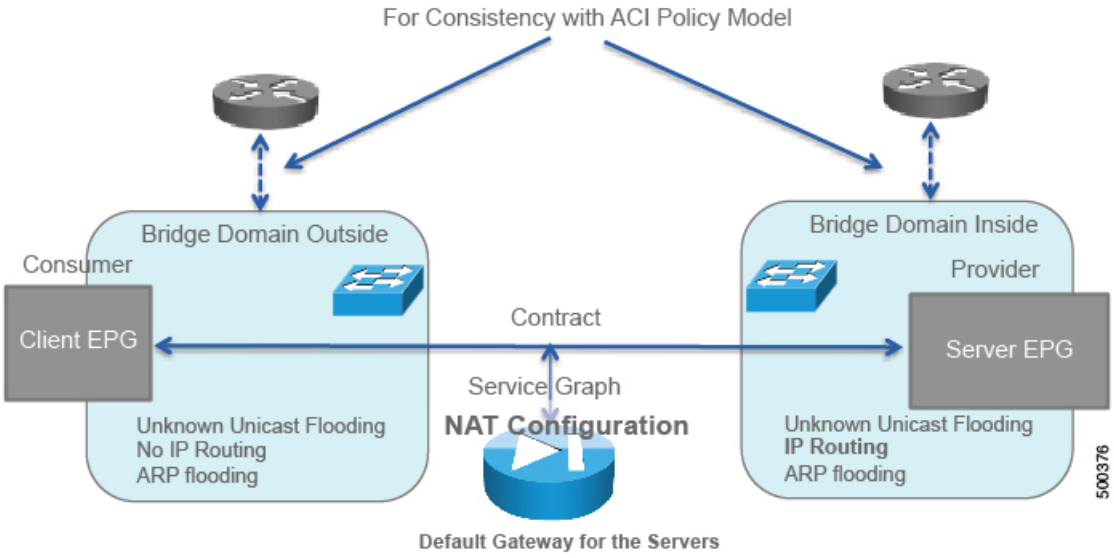
When you configure the bridge domains for ASA in GoTo mode, configure the bridge domains as you would for a generic configuration, except as follows:

- **L2 Unknown Unicast** radio buttons—Choose **Flood**.
- **ARP Flooding** check box—Put a check in the check box.
- **Unicast Routing** check box—Put a check in the check box if you need to configure an L3Out or for the endpoint attach feature.

For information on how to configure bridge domains, see Creating Bridge Domains and VRFs Using the GUI, on page 36.

The following figure illustrates the bridge domain configuration for ASA in GoTo mode:

***Figure 56: Bridge Domain Configuration for ASA in GoTo Mode***




If you need the mapping database, such as for using traceroute or endpoint attach, you must enable unicast routing in the bridge domains.

# Tuning the Server-Side Bridge Domain for Flood Removal for ASA in GoTo Mode

In GoTo mode you might want to optimize flooding. This tuning is meaningful only in the case of a service graph with GoTo mode, because in GoThrough mode Cisco Application Centric Infrastructure (ACI) sets the bridge domains to unknown unicast flooding.

On the server-side bridge domain, it can be beneficial to reduce flooding for unknown unicast packets. To do this, you can enable hardware proxy on the bridge domain. You should keep ARP flooding enabled because it might be necessary in the presence of ASA deployed in HA pairs.

# Adding Endpoint Attach Support for ASA in GoTo Mode

You can deploy an ASA device in a service graph in a way that the endpoints that are discovered in the provider endpoint group are automatically added to an object group. In the ASA device, this feature is called "endpoint attach".

The object group is given a name in the following format:

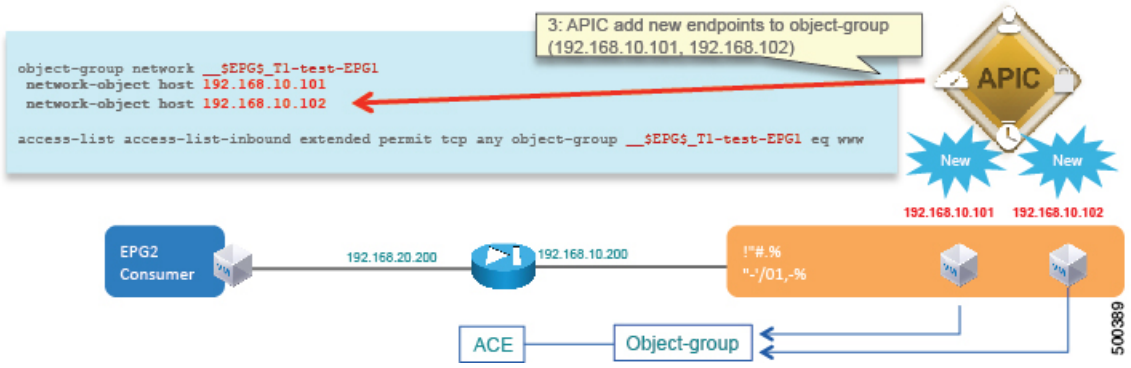
*tenant_name-application_profile_name-EPG_name*

For example:

```
ciscoasa# show run object-group
object-group network __$EPG$_T1-test-EPG1
  network-object host 192.168.10.100
```

The ACL will then reference this object group. The endpoint group detects the endpoint and populates the ACL. The Application Policy Infrastructure Controller (APIC) dynamically detects the new endpoint, then the endpoint is automatically added to the object group for ACE.

***Figure 57: Example of the APIC Adding New Endpoints to the Object Group***




The following procedure enables endpoint attach.

### Procedure

**Step 1** On the menu bar, choose **Tenants** > **All Tenants**.

**Step 2** In the Work pane, double click the tenant's name.

**Step 3** In the Navigation pane, choose **Tenant** *tenant_name* > **L4-L7 Services** > **L4-L7 Service Graph Template** > *service_graph_template_name* > **Function Node -** *node_name* > **provider**.

**Step 4** In the Work pane, choose the connector's properties.

**Step 5** Put a check in the **Attachment Notification** check box.

**Step 6** Click **Submit**.

**Step 7** Configure Layer 4 to Layer 7 parameters for ASA. The bridge domain must have routing enabled.

**Step 8** Configure the ACE by defining the Device Config > Access List > Access Control Entry > Destination Address > Endpoint Group parameter as `epg_name` and set the value equal to the object group name in the following format:

*tenant_name-application_profile_name-EPG_name*

For example:

```
<!-- destination address: any or dynamically populated -->
<vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="FW-routed" key="destination_address"
 name="dest-address" nodeNameOrLbl="ASA-1-node" >
    <!-- destination address: autopopulated from the EPG endpoints -->
    <!-- Format is Tenant-applicationprofile-EPG -->
    <vnsParamInst  key="epg_name"   name="epg_name"  value="Sales-orderingtool-app"/>
    <!-- destination address: any -->
    <!-- vnsParamInst key="any" name="any"  value="any" -->
</vnsFolderInst>
```

# ASA GoTo Mode Design Examples

The following figures illustrate ASA GoTo mode deployments with various scenarios: some with the client connected directly to the fabric, some with the fabric providing routing to the outside, and some with an external router. The figures include the recommended bridge domain settings for both client and server-side bridge domains.

The settings for the server-side or provider-side (also known as the internal bridge domain, `BD2`) include IP routing in case you decide to use the endpoint attach feature. If you do not want to use endpoint attach and

you do not care about flood reduction in the server-side bridge domain, you can configure the bridge domain without IP routing.

***Figure 58: GoTo Mode with Client VMs (Split VRF)***

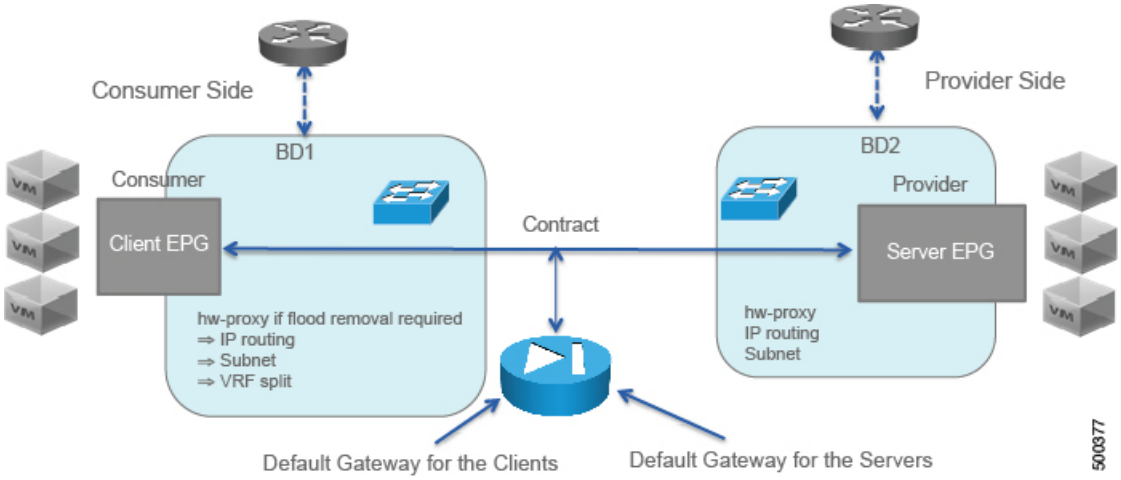



***Figure 59: GoTo Mode with L3out option 1 with NAT and a single VRF)***

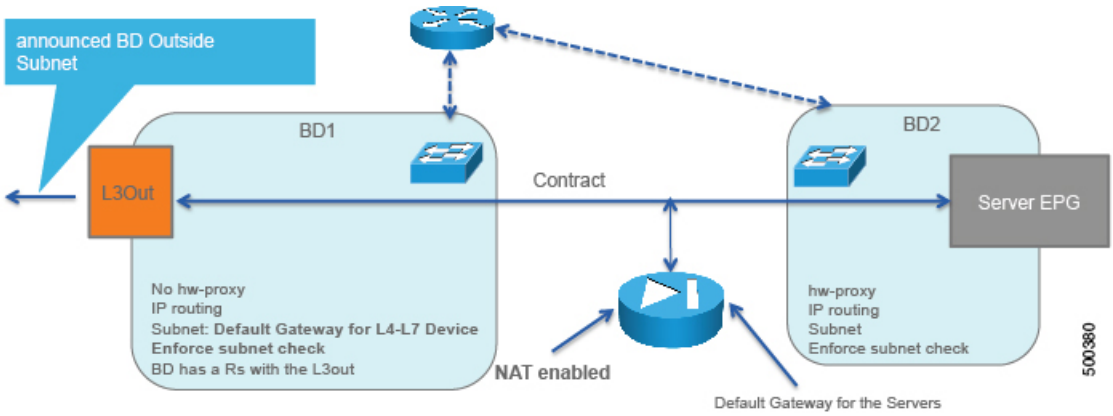



***Figure 60: GoTo Mode Using Two VRFs***

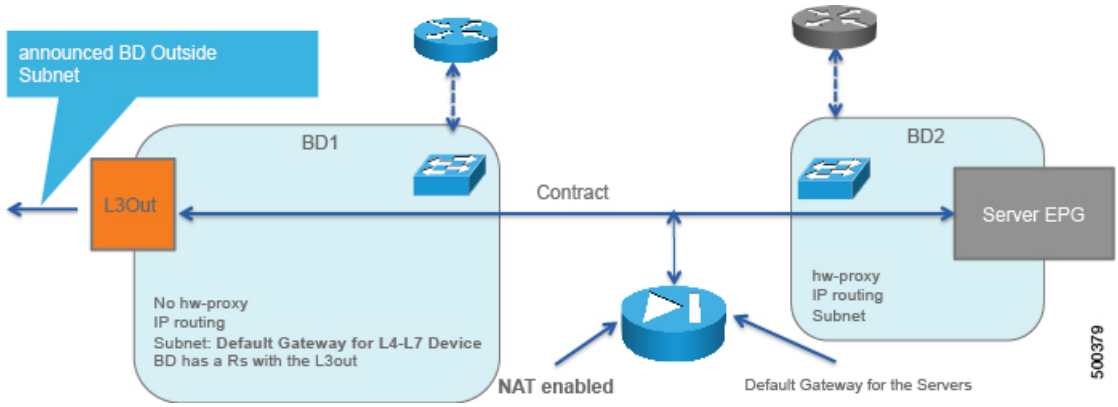
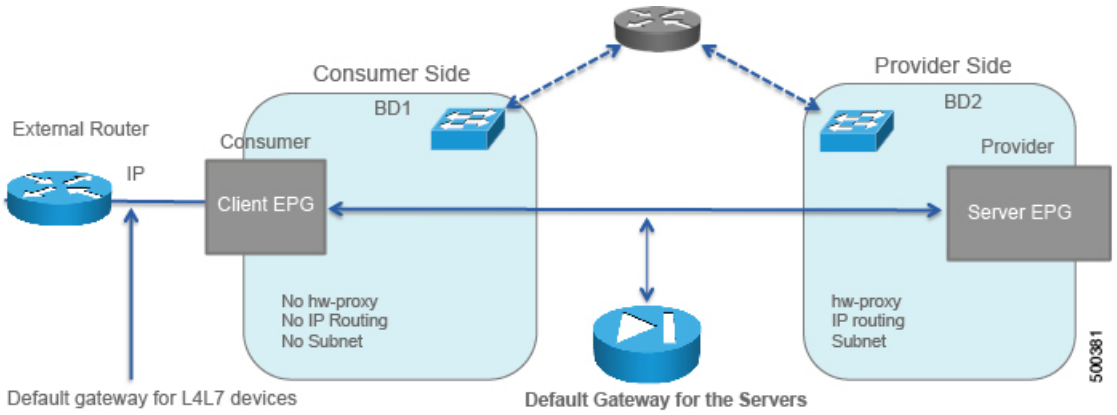



*Figure 61: GoTo Mode with External Router*




# Deploying ASA in GoTo Mode

The tasks that you must perform to deploy ASA in GoTo mode are nearly identical to the tasks for generically deploying a service graph, with a few differences. The following procedure provides the generic service graph deployment tasks, along with information about what you must do differently to deploy ASA in GoTo mode.

### Procedure

**Step 1** Import the device package.
See Importing a Device Package Using the GUI, on page 35.

**Step 2** Create the bridge domains and VRFs.
See Creating Bridge Domains and VRFs Using the GUI, on page 36.

a) For the inside bridge domain, enable **Unicast Routing** if you plan to use endpoint attach.
b) Associate the bridge domain with a VRF, which is necessary because of the object model. The hardware will not program the VRF if the bridge domain is configured only as Layer 2.

**Step 3** Create endpoint groups and contracts.
See Creating Endpoint Groups and Contracts Using the GUI, on page 37.

**Step 4** Configure logical devices and concrete devices.
See Creating a Logical or Concrete Device Using the GUI, on page 39.

a) For a concrete device, in the **Service Type** drop-down list, choose **Firewall**.
b) For the **Function Type** buttons, click **GoTo**.
c) For the Layer 4 to Layer 7 parameters, for the Host Name parameter, set the value to the host name of the ASA device.

If you have not yet applied the service graph template, a concrete device will have a health score of 0. This indicates the vNICs are not yet connected to a valid port group, which is normal since the graph has not been applied yet. As long as the device has a **Device State** of `stable`, then the communication between Application Policy Infrastructure Controller (APIC) and the device is working.

**Step 5** Create or import a function profile.
See Creating a Function Profile Using the GUI, on page 43 or Importing a Function Profile Using the GUI, on page 44.

- The configuration parameters for the firewall at the `CDev` level include the port channel, but they do not include the IP address. The reason is that the IP address of the firewall can change depending on where it is deployed, such as in which graph or tenant it is deployed.
- In this configuration, you must configure the device parameters for the port channel by using the "ALL parameters" field and set the LACP maximum to "8".
- You need to define each LACP member in the parameters.
- The VLAN on the port channel is automatically created in the rendering phase based on the bridge domain information and based on the physical domain information.

The following table describes the mandatory Layer 4 to Layer 7 parameters and provides examples of possible values that you must change for your specific configuration:

***Table 3: Layer 4 to Layer 7 Parameters for ASA in GoTo Mode***

| L4-L7 Parameter or Folder | ASA Usage and Notes |
|---|---|
| `Device Config` folder | Define as `Device`. |
| `Device Config > Access List` folder | Define as `access-list-inbound`. |
| `Device Config > Access List > Access Control Entry` folder | Define as `permit-icmp`.<br>Expand this folder to enter the Application Control Engine (ACE) parameters. |
| `Device Config > Access List > Access Control Entry` folder | Define as `permit-ssh`.<br>Expand this folder to enter the ACE parameters. |
| `Device Config > NAT Rules List` folder | Define as `NATList-A`. |

| L4-L7 Parameter or Folder | ASA Usage and Notes |
|---|---|
| `Device Config > NAT Rules List > NAT Rule` folder | Define as `NATRule1`. |
| `Device Config > NAT Rules List > NAT Rule > Destination Address Translation > Mapped Object > Network Object` parameter | Define as `object_name` with a value of `Server1OutsideIP`. This is a traffic selection object. |
| `Device Config > NAT Rules List > NAT Rule > Destination Address Translation > Real Object > Network Object` parameter | Define as `object_name` with a value of `Server1InsideIP`. This is a traffic selection object. |
| `Device Config > Network Object` folder | Define as `ServerInsideP`. |
| `Device Config > Network Object > Host IP Address` parameter | Define as `host_ip_address` with a value of `30.0.0.101`. |
| `Device Config > Network Object` folder | Define as `ServerOutsideP`. |
| `Device Config > Network Object > Host IP Address` parameter | Define as `host_ip_address` with a value of `10.0.0.11`. |
| `Interface Related Configuration` folder for externalIf | Define as `externalIf`. |
| `Interface Related Configuration > Access Group` folder | Define as `ExtAccessGroup`. |
| `Interface Related Configuration > Access Group > Inbound Access List` parameter | Define as `name` with a value of `access-list-inbound`. |

| L4-L7 Parameter or Folder | ASA Usage and Notes |
|---|---|
| `Interface Related Configuration > Interface Specific Configuration` folder | Define as `externalIfCfg`. |
| `Interface Related Configuration > Interface Specific Configuration > IPv4 Address Configuration` folder | Define as `IPv4Address`. |
| `Interface Related Configuration > Interface Specific Configuration > IPv4 Address Configuration > IPv4 Address` parameter | Define as `ipv4_address` with a value of `10.0.0.3/255.255.255.0`. This mask value must follow this exact format. |
| `Interface Related Configuration` folder for internalIf | Define as `internalIf`. |
| `Interface Related Configuration > Interface Specific Configuration` folder | Define as `internalIfCfg`. |
| `Interface Related Configuration > Interface Specific Configuration > IPv4 Address Configuration` folder | Define as `IPv4Address`. |
| `Interface Related Configuration > Interface Specific Configuration > IPv4 Address Configuration > IPv4 Address` parameter | Define as `ipv4_address` with a value of `30.0.0.3/255.255.255.0`. This mask value must follow this exact format.<br>The internalIf does not require parameters for an ACL. |

| L4-L7 Parameter or Folder | ASA Usage and Notes |
|---|---|
| `Function Config` folder | Define as `Function`. From this folder, you must reference the interfaces and the NAT configuration.<br>In this folder, do the following things:<br>1 Define the `NAT Policy` folder as `NATPolicy`.<br>2 Define the NAT Policy > NAT Rules List parameter as `nat_list_name` with a value of `NATList-A`. |
| `Function Config > NAT Policy` folder | Define as `NATPolicy`. |
| `Function Config > NAT Policy > NAT Rules List` parameter | Define as `nat_list_name` with a value of `NATList-A`. |

The following XML illustrates an example of Layer 4 to Layer 7 parameters configuration for the ASA deployment in GoTo mode:

```
<!-- RELATION TO THE EXTERNAL AND INTERNAL INTERFACES  -->
<vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="FW-routed"
key="ExIntfConfigRelFolder" name="ExtConfig" nodeNameOrLbl="ASA-1-node" >
    <vnsCfgRelInst key="ExIntfConfigRel" name="ExtConfigrel" targetName="externalIf"/>
</vnsFolderInst>
<vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="FW-routed"
key="InIntfConfigRelFolder" name="IntConfig" nodeNameOrLbl="ASA-1-node" >
    <vnsCfgRelInst key="InIntfConfigRel" name="InConfigrel" targetName="internalIf"/>
</vnsFolderInst>

<!-- ACL DEFINITION, ACL NAME "access-list-inbound" -->
<vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="FW-routed" key="AccessList"
name="access-list-inbound" nodeNameOrLbl="ASA-1-node" >

<!-- ACE "permit-ssh" -->
    <vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="FW-routed"
key="AccessControlEntry" name="permit-ssh" nodeNameOrLbl="ASA-1-node" >
        <vnsParamInst key="order" name="order1"  value="10"/>
        <!-- protocol -->
        <vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="FW-routed" key="protocol"
 name="tcp" nodeNameOrLbl="ASA-1-node" >
            <vnsParamInst key="name_number" name="tcp"  value="tcp"/>
        </vnsFolderInst>
        <!-- source address -->
        <vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="FW-routed"
key="source_address" name="src-address" nodeNameOrLbl="ASA-1-node" >
            <vnsParamInst key="any" name="any"  value="any"/>
        </vnsFolderInst>
        <!-- destination address -->
        <vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="FW-routed"
key="destination_address" name="dest-address" nodeNameOrLbl="ASA-1-node" >
```

```
            <vnsParamInst key="any" name="any"  value="any"/>
        </vnsFolderInst>
        <!-- destination L4 port -->
        <vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="FW-routed"
key="destination_service" name="dest-service" nodeNameOrLbl="ASA-1-node" >
            <vnsParamInst key="operator" name="op"  value="eq"/>
            <vnsParamInst key="low_port" name="port"  value="22"/>
        </vnsFolderInst>
        <!-- action permit or deny -->
        <vnsParamInst key="action" name="action-permit"  value="permit"/>
    </vnsFolderInst>
    <!-- ACE "permit-icmp" -->
    <vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="FW-routed"
key="AccessControlEntry" name="permit-icmp" nodeNameOrLbl="ASA-1-node" >
        <vnsParamInst key="order" name="order1"  value="10"/>
        <!-- protocol -->
        <vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="FW-routed" key="protocol"
 name="icmp" nodeNameOrLbl="ASA-1-node" >
            <vnsParamInst key="name_number" name="icmp"  value="icmp"/>
        </vnsFolderInst>
        <!-- source address -->
        <vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="FW-routed"
key="source_address" name="src-address" nodeNameOrLbl="ASA-1-node" >
            <vnsParamInst key="any" name="any"  value="any"/>
        </vnsFolderInst>
        <!-- destination address -->
        <vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="FW-routed"
key="destination_address" name="dest-address" nodeNameOrLbl="ASA-1-node" >
            <vnsParamInst key="any" name="any"  value="any"/>
        </vnsFolderInst>
        <!-- action -->
        <vnsParamInst key="action" name="action-permit"  value="permit"/>
    </vnsFolderInst>
</vnsFolderInst>

<!-- EXTERNAL INTERFACE  -->
<vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="FW-routed" key="Interface"
name="externalIf" nodeNameOrLbl="ASA-1-node" >
   <vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="FW-routed" key="InterfaceConfig"
 name="externalIfCfg" nodeNameOrLbl="ASA-1-node" >
        <!-- security level -->
        <vnsParamInst key="security_level" name="external_security_level"  value="50"/>
        <!-- IP ADDRESS-->
       <vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="FW-routed" key="IPv4Address"
 name="IPv4Address" nodeNameOrLbl="ASA-1-node" >
            <vnsParamInst key="ipv4_address" name="ipv4_address"
value="10.0.0.3/255.255.255.0"/>
        </vnsFolderInst>
    </vnsFolderInst>
    <!-- access-group  -->
    <vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="FW-routed" key="AccessGroup"
name="ExtAccessGroup" nodeNameOrLbl="ASA-1-node" >
        <vnsCfgRelInst key="inbound_access_list_name" name="name"
targetName="access-list-inbound"/>
    </vnsFolderInst>
```

```
</vnsFolderInst>
<vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="FW-routed" key="Interface"
name="internalIf" nodeNameOrLbl="ASA-1-node" >
   <vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="FW-routed" key="InterfaceConfig"
 name="internalIfCfg" nodeNameOrLbl="ASA-1-node" >
        <!-- security level -->
        <vnsParamInst key="security_level" name="internal_security_level"  value="100"/>
        <!-- IP ADDRESS-->
       <vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="FW-routed" key="IPv4Address"
 name="IPv4Address" nodeNameOrLbl="ASA-1-node" >
            <vnsParamInst key="ipv4_address" name="ipv4_address"
value="30.0.0.3/255.255.255.0"/>
        </vnsFolderInst>
    </vnsFolderInst>
</vnsFolderInst>
```

**Step 6** Create a service graph template and either use a function profile or enter the Layer 4 to Layer 7 parameters by hand.
See Creating a Layer 4 to Layer 7 Service Graph Template Using the GUI, on page 44.

a) Drag the defined logical device to the canvas.

**Step 7** Apply the service graph template.
See Applying a Service Graph Template to Endpoint Groups Using the GUI, on page 45.

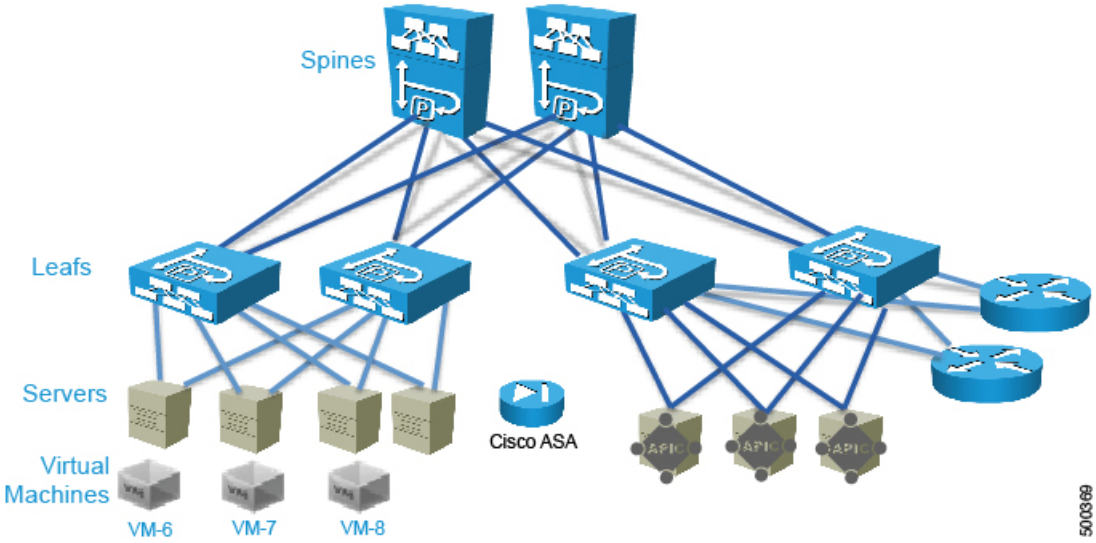
**Step 8** Verify that the configuration deployed successfully.
See Verifying the Configuration for an ASA Device, on page 105.

# ASA in GoThrough Mode

## About Deploying ASA in GoThrough Mode

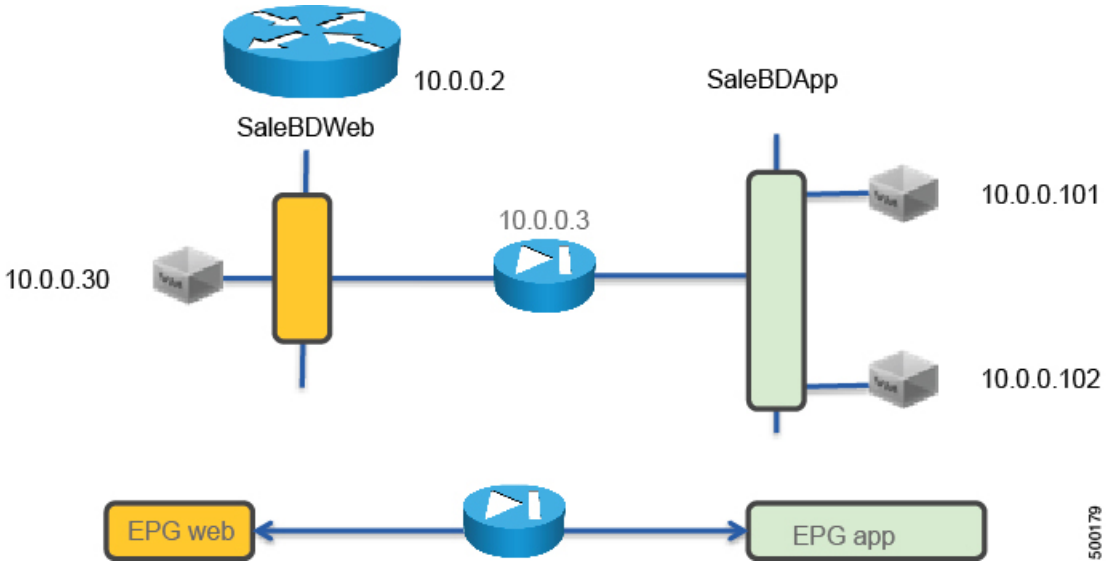
The following figure illustrates the topology for deploying Cisco Application Centric Infrastructure (ACI) fabric with ASA devices:

***Figure 62: ACI Fabric with ASA Devices***




The following figure illustrates the logical topology of an ASA GoThrough deployment:

***Figure 63: Logical Topology of an ASA GoThrough Deployment***



To deploy an ASA device in the GoThrough mode, you must do the following things:

- Configure 2 bridge domains
- Configure 2 endpoint groups with each one associated with a different bridge domain
- Enable routing on only one of the two bridge domains, which normally would be the outside bridge domain for the purpose of an L3Out
- Enable ARP flooding and unknown unicast flooding on both bridge domains
- Configure the ASA device as a GoThrough device
- Configure the contract between the outside and inside endpoint group (or server side or provider side)
- Associate the service graph with the contract
- Associate the external logical interface with GigabitEthernet0/0 (which in the case of ASAv is Network Adapter 2)
- Associate the internal logical interface with GigabitEthernet0/1 (which in the case of ASAv is Network Adapter 3)

# Overview of Preparing an ASA Device in GoThrough Mode

ASA and ASAv do not have the concept of VRF management. For GoThrough mode, you do not need to use "inband" management because there is no need for VRF management.

The following procedure provides of overview of preparing an ASA device to be deployed in GoThrough mode.

### Procedure

**Step 1** Enable SSH.

**Step 2** Enable HTTP access.

**Step 3** Configure the credentials.
You do not need to configure the interfaces, VLANs, or IP addresses.

**Step 4** Enter the following commands to create the initial configuration:

```
asa1(config)# firewall transparent
asa1(config)# Interface Management0/0
asa1(config)# nameif management
asa1(config)# ip address 192.168.12.120 255.255.255
asa1(config)# no shut
asa1(config)# hostname ASAv
asa1(config)# route management 0.0.0.0 0.0.0.0 192.168.12.254
asa1(config)# user admin password tme12345
asa1(config)# enable password tme12345
asa1(config)# aaa authentication ssh console LOCAL
asa1(config)# http server enable
asa1(config)# http 0.0.0.0 0.0.0.0 management
asa1(config)# ssh 0.0.0.0 0.0.0.0 management
```

# Configuring Bridge Domains for ASA in GoThrough Mode

While you can optimize the bridge domain settings for GoThrough mode and optimize flooding, in practice the GoThrough service graph modifies the bridge domains to enable unknown unicast flooding and ARP flooding. ARP flooding is needed to make sure that if a firewall changes its MAC address while keeping the same IP address as a result of a failover, the gratuitous ARP can reach all the servers in the bridge domain to get updated to point to the new MAC address. With ARP flooding enabled, hypothetically unknown unicast flooding would not be needed, but it is assumed that the firewall relies on flooding to discover where each MAC address is and to build the forwarding table.

IP routing can be enabled on both bridge domains if each bridge domain had a different VRF. However, the service graph for GoThrough mode changes the bridge domain settings and does not render if both bridge domains have IP routing enabled.

When you configure the bridge domains for ASA in GoThrough mode, configure the bridge domains as you would for a generic configuration, except as follows:

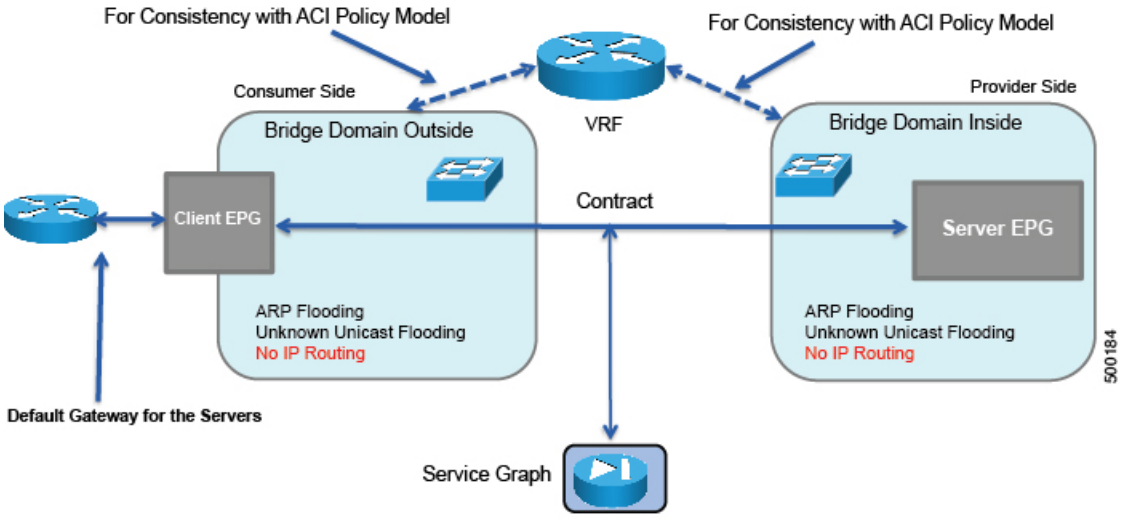
- **L2 Unknown Unicast** radio buttons—Choose **Flood**.
- **ARP Flooding** check box—Put a check in the check box.
- **Unicast Routing** check box—Put a check in the check box.
- **Unicast Routing** check box—Put a check in the check box if you are configuring the outside bridge domain and the Cisco Application Centric Infrastructure (ACI) fabric is the default gateway for the servers.

The GoThrough mode service graph does not render if IP routing is enabled on both bridge domains, and endpoint attach is not designed to work with GoThrough mode.

For information on how to configure bridge domains, see Creating Bridge Domains and VRFs Using the GUI, on page 36.

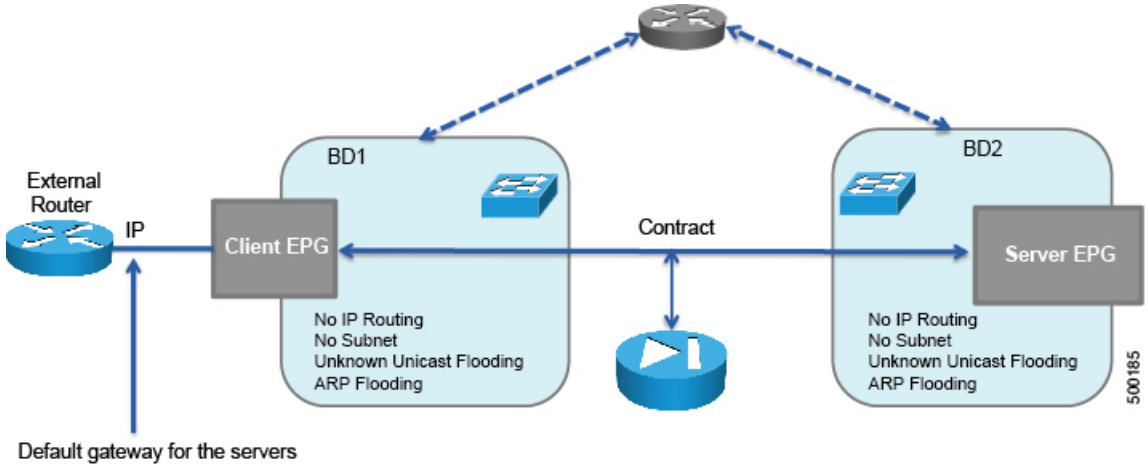
The following figure illustrates the simplest bridge domain configuration for ASA in GoThrough mode:

***Figure 64: Simplest Bridge Domain Configuration for ASA in GoThrough Mode***



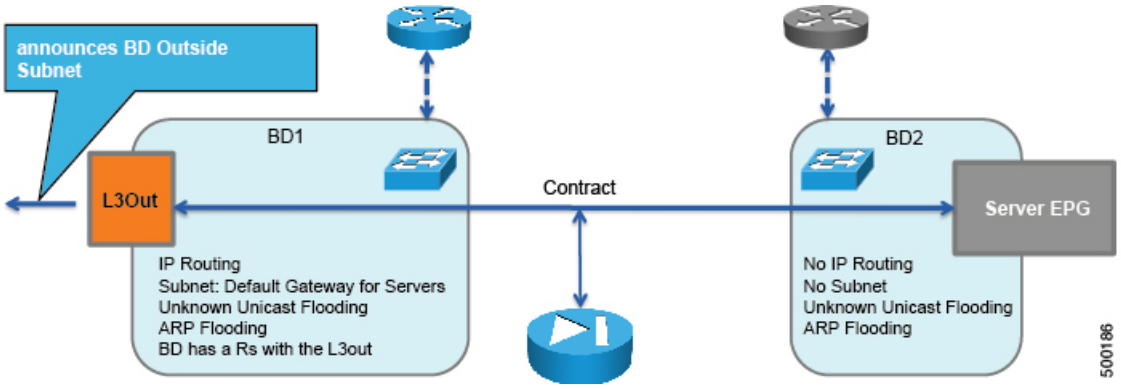
The following figure illustrates the bridge domain configuration for ASA deployment in GoThrough mode with an external router:

***Figure 65: Bridge Domain Configuration for ASA in GoThrough Mode with an External Router***




The following figure illustrates the bridge domain configuration for ASA deployment in GoThrough mode with an L3Out:

***Figure 66: Bridge Domain Configuration for ASA in GoThrough Mode with an L3Out***




# Deploying ASA in GoThrough Mode

The tasks that you must perform to deploy ASA in GoThrough mode are nearly identical to the tasks for generically deploying a service graph, with a few differences. The following procedure provides the generic service graph deployment tasks, along with information about what you must do differently to deploy ASA in GoThrough mode.

### Procedure

**Step 1** Import the device package.

See Importing a Device Package Using the GUI, on page 35.

**Step 2** Create the bridge domains and VRFs.
See Creating Bridge Domains and VRFs Using the GUI, on page 36.

a) Associate the bridge domain with a VRF, which is necessary because of the object model. The hardware will not program the VRF if the bridge domain is configured only as Layer 2.

**Step 3** Create endpoint groups and contracts.
See Creating Endpoint Groups and Contracts Using the GUI, on page 37.

**Step 4** Configure logical devices and concrete devices.
See Creating a Logical or Concrete Device Using the GUI, on page 39.

a) For a concrete device, in the **Service Type** drop-down list, choose **Firewall**.
b) For the **Function Type** buttons, click **GoThrough**.
c) For the Layer 4 to Layer 7 parameters, for the Host Name parameter, set the value to the host name of the ASA device.

If you have not yet applied the service graph template, a concrete device will have a health score of 0. This indicates the vNICs are not yet connected to a valid port group, which is normal since the graph has not been applied yet. As long as the device has a **Device State** of `stable`, then the communication between Application Policy Infrastructure Controller (APIC) and the device is working.

**Step 5** Create or import a function profile.
See Creating a Function Profile Using the GUI, on page 43 or Importing a Function Profile Using the GUI, on page 44.

- The configuration parameters for the firewall at the `CDev` level include the port channel, but they do not include the IP address. The reason is that the IP address of the firewall can change depending on where it is deployed, such as in which graph or tenant it is deployed.
- In this configuration, you must configure the device parameters for the port channel by using the "ALL parameters" field and set the LACP maximum to "8".
- You need to define each LACP member in the parameters.
- The VLAN on the port channel is automatically created in the rendering phase based on the bridge domain information and based on the physical domain information.

The following table describes the mandatory Layer 4 to Layer 7 parameters and provides examples of possible values that you must change for your specific configuration:

***Table 4: Layer 4 to Layer 7 Parameters for ASA in GoThrough Mode***

| L4-L7 Parameter or Folder | Usage and Notes |
|---|---|
| `Device Config` folder | Define as `Device`. |
| `Device Config > Access List > Access Control Entry` folder | Define as `permit-icmp`.<br>Expand this folder to enter the Application Control Engine (ACE) parameters. |

| L4-L7 Parameter or Folder | Usage and Notes |
|---|---|
| `Device Config > Access List > Access Control Entry` folder | Define as `permit-ssh`.<br>Expand this folder to enter the ACE parameters. |
| `Device Config > Bridge Group Interface` folder | Define as `1`. |
| `Device Config > Bridge Group Interface > IPv4 Address Configuration > IPv4 Address` parameter | Define as `ipv4_address` with a value in the following format:<br>*a.b.c.d*/*e.f.g.h*<br>*a.b.c.d* is the IPv4 address, while *e.f.g.h* is the mask. For example:<br>`10.0.0.2/255.255.255.0`<br>. |
| `Interface Related Configuration` folder for externalIf | Define as `externalIf`. |
| `Interface Related Configuration > Access Group` folder | Define as `ExtAccessGroup`. |
| `Interface Related Configuration > Interface Specific Configuration` folder | Define as `externalIfCfg`. |
| `Interface Related Configuration > Interface Specific Configuration > Bridge Group Interface` parameter | Define as `extbridge` with a value of the bridge group number that you defined for the `Device Config > Bridge Group Interface` folder. |
| `Interface Related Configuration` folder for internalIf | Define as `internalIf`.<br>The internalIf does not require parameters for an ACL. |
| `Interface Related Configuration > Interface Specific Configuration` folder | Define as `internalIfCfg`. |
| `Interface Related Configuration > Interface Specific Configuration > Bridge Group Interface` parameter | Define as `intbridge` with a value of the bridge group number that you defined for the `Device Config > Bridge Group Interface` folder. |

The following XML is an example of a Layer 4 to Layer 7 parameters configuration:

```
<!-- RELATION TO THE EXTERNAL AND INTERNAL INTERFACES  -->
<vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="FW-bridged"
key="ExIntfConfigRelFolder" name="ExtConfig" nodeNameOrLbl="ASA-1-node" >
    <vnsCfgRelInst key="ExIntfConfigRel" name="ExtConfigrel" targetName="externalIf"/>
</vnsFolderInst>
<vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="FW-bridged"
key="InIntfConfigRelFolder" name="IntConfig" nodeNameOrLbl="ASA-1-node" >
    <vnsCfgRelInst key="InIntfConfigRel" name="InConfigrel" targetName="internalIf"/>
</vnsFolderInst>

<!-- ACL DEFINITION, ACL NAME "access-list-inbound" -->
<vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="FW-bridged" key="AccessList"
name="access-list-inbound" nodeNameOrLbl="ASA-1-node" >

<!-- ACE "permit-ssh" -->
    <vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="FW-bridged"
key="AccessControlEntry" name="permit-ssh" nodeNameOrLbl="ASA-1-node" >
        <vnsParamInst key="order" name="order1"  value="10"/>
        <!-- protocol -->
       <vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="FW-bridged" key="protocol"
 name="tcp" nodeNameOrLbl="ASA-1-node" >
            <vnsParamInst key="name_number" name="tcp"  value="tcp"/>
        </vnsFolderInst>
        <!-- source address -->
        <vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="FW-bridged"
key="source_address" name="src-address" nodeNameOrLbl="ASA-1-node" >
            <vnsParamInst key="any" name="any"  value="any"/>
        </vnsFolderInst>
        <!-- destination address -->
        <vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="FW-bridged"
key="destination_address" name="dest-address" nodeNameOrLbl="ASA-1-node" >
            <vnsParamInst key="any" name="any"  value="any"/>
        </vnsFolderInst>
        <!-- destination L4 port -->
        <vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="FW-bridged"
key="destination_service" name="dest-service" nodeNameOrLbl="ASA-1-node" >
            <vnsParamInst key="operator" name="op"  value="eq"/>
            <vnsParamInst key="low_port" name="port"  value="22"/>
        </vnsFolderInst>
        <!-- action permit or deny -->
        <vnsParamInst key="action" name="action-permit"  value="permit"/>
    </vnsFolderInst>
    <!-- ACE "permit-icmp" -->
    <vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="FW-bridged"
key="AccessControlEntry" name="permit-icmp" nodeNameOrLbl="ASA-1-node" >
        <vnsParamInst key="order" name="order1"  value="10"/>
        <!-- protocol -->
       <vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="FW-bridged" key="protocol"
 name="icmp" nodeNameOrLbl="ASA-1-node" >
            <vnsParamInst key="name_number" name="icmp"  value="icmp"/>
        </vnsFolderInst>
```

```
        <!-- source address -->
        <vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="FW-bridged"
key="source_address" name="src-address" nodeNameOrLbl="ASA-1-node" >
            <vnsParamInst key="any" name="any"  value="any"/>
        </vnsFolderInst>
        <!-- destination address -->
        <vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="FW-bridged"
key="destination_address" name="dest-address" nodeNameOrLbl="ASA-1-node" >
            <vnsParamInst key="any" name="any"  value="any"/>
        </vnsFolderInst>
        <!-- action -->
        <vnsParamInst key="action" name="action-permit"  value="permit"/>
    </vnsFolderInst>
</vnsFolderInst>

<!-- BRIDGE-GROUP 1 -->
<vnsFolderInst  ctrctNameOrLbl="webtoapp"  graphNameOrLbl="FW-bridged" key="BridgeGroupIntf"
 name="1" nodeNameOrLbl="ASA-1-node" scopedBy="epg">
    <vnsParamInst  key="ipv6_nd_dad_attempts"  name="ipv6_nd_dad_attempts" validation=""
value="1"/>
    <!-- IP ADDRESS-->
   <vnsFolderInst  ctrctNameOrLbl="webtoapp"  graphNameOrLbl="FW-bridged" key="IPv4Address"
 name="IPv4Address" nodeNameOrLbl="ASA-1-node" scopedBy="epg">
        <vnsParamInst  key="ipv4_address"  name="ipv4_address" validation=""
value="30.0.0.254/255.255.255.0"/>
    </vnsFolderInst>
</vnsFolderInst>

<!-- EXTERNAL INTERFACE  -->
<vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="FW-bridged" key="Interface"
name="externalIf" nodeNameOrLbl="ASA-1-node" >
   <vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="FW-bridged" key="InterfaceConfig"
 name="externalIfCfg" nodeNameOrLbl="ASA-1-node" >
        <!-- BRIDGE-GROUP CONFIGURATION -->
        <vnsCfgRelInst  key="bridge_group"  name="extbridge" targetName="1"/>
        <!-- security level -->
        <vnsParamInst key="security_level" name="external_security_level"  value="50"/>
    </vnsFolderInst>
    <!-- access-group  -->
    <vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="FW-bridged" key="AccessGroup"
 name="ExtAccessGroup" nodeNameOrLbl="ASA-1-node" >
        <vnsCfgRelInst key="inbound_access_list_name" name="name"
targetName="access-list-inbound"/>
    </vnsFolderInst>
</vnsFolderInst>
<vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="FW-bridged" key="Interface"
name="internalIf" nodeNameOrLbl="ASA-1-node" >
   <vnsFolderInst ctrctNameOrLbl="webtoapp" graphNameOrLbl="FW-bridged" key="InterfaceConfig"
 name="internalIfCfg" nodeNameOrLbl="ASA-1-node" >
        <!-- BRIDGE-GROUP CONFIGURATION -->
        <vnsCfgRelInst  key="bridge_group"  name="intbridge" targetName="1"/>
        <!-- security level -->
        <vnsParamInst key="security_level" name="internal_security_level"  value="100"/>
    </vnsFolderInst>
</vnsFolderInst>
```

**Step 6** Create a service graph template and either use a function profile or enter the Layer 4 to Layer 7 parameters by hand.
See Creating a Layer 4 to Layer 7 Service Graph Template Using the GUI, on page 44.

a) Drag the defined logical device to the canvas.
b) In the **ASA Cluster Information** section, for the **Firewall** radio buttons, choose **Two-Arm**.

**Step 7** Apply the service graph template.
See Applying a Service Graph Template to Endpoint Groups Using the GUI, on page 45.

You cannot configure an "any" virtual IP or port. You can only choose **TCP** or **UDP** option; there is no "all IP protocol" value.

**Step 8** Verify that the configuration deployed successfully.
See Verifying the Configuration for an ASA Device, on page 105.

# Verifying the Configuration for an ASA Device

After you deployed an ASA device in any mode, you can verify that the configuration is functioning properly by using the following procedure. If you encounter an issue, you can try troubleshooting by viewing the Application Policy Infrastructure Controller (APIC) log that is at the following location:

```
/data/devicescript/CISCO.ASA.1.2/logs/debug.log
```

### Procedure

**Step 1** In the APIC GUI, on the menu bar, choose **Tenants** > **All Tenants**.

**Step 2** In the Work pane, double click the tenant's name.

**Step 3** In the Navigation pane, choose **Tenant** ***tenant_name*** > **L4-L7 Services** > **Deployed Graph Instances** > ***ASA_graph_name***.

**Step 4** In the Work pane, in the **Cluster Interfaces** section, ensure that the logical interfaces appear.

**Step 5** In the Navigation pane, choose **Tenant** ***tenant_name*** > **L4-L7 Services** > **Deployed Devices** > ***ASA_device_name***.

**Step 6** In the Work pane, view the ASA device's properties. The health score should be 100.

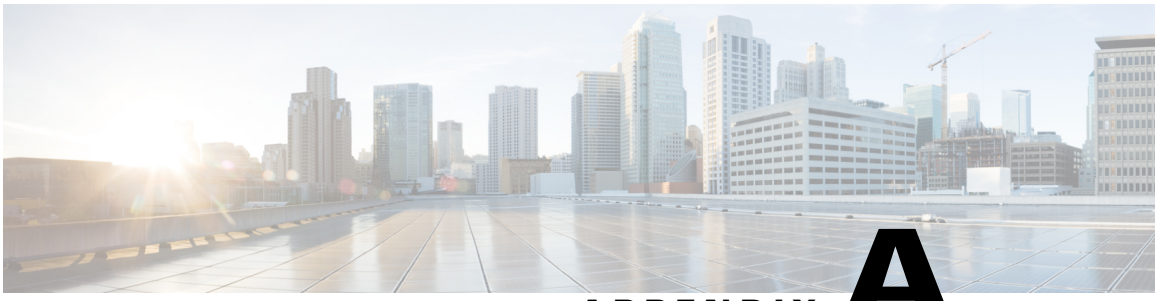
**Step 7** In the ASA GUI, choose the **Virtual Hardware** tab.
Verify that the vNICs were automatically placed in the shadow EPGs.

**Step 8** In the Cisco Adaptive Security Device Manager (ASDM) GUI, choose **Configuration** > **Device Setup** > **Interface Settings** > **Interfaces**.
In the Work pane, verify that you can see the `externalIf` and `internalIf` interfaces.

# Undoing a Service Graph Configuration for ASA

To undo a service graph configuration for ASA, in the Application Policy Infrastructure Controller (APIC) GUI, delete the service graph template.

See Undoing a Service Graph Configuration Using the GUI, on page 49.



APPENDIX A

# Route Peering

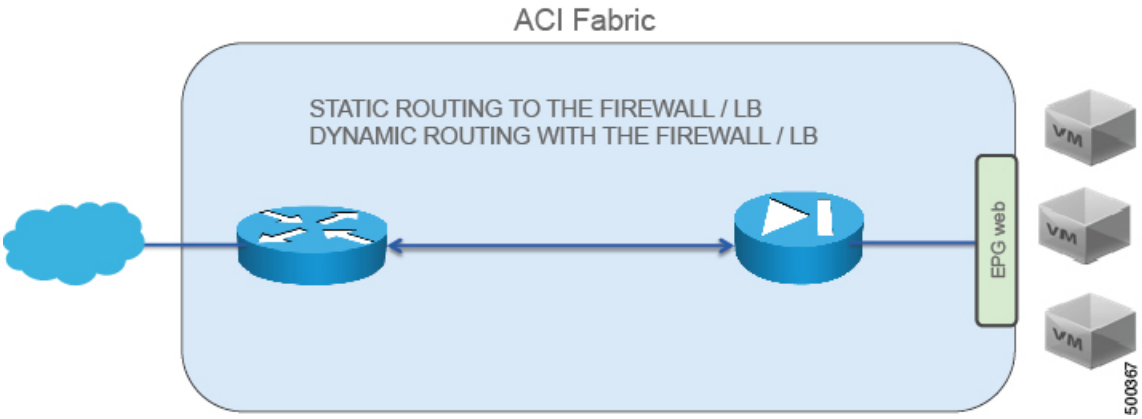


## About Route Peering

Route peering is a special case of the more generic Cisco Application Centric Infrastructure (ACI) fabric as a transit use case, in which route peering enables the ACI fabric to serve as a transit domain for Open Shortest Path First (OSPF) or Border Gateway Protocol (BGP) protocols. A common use case for route peering is route health injection, in which the server load balancing virtual IP is advertised over OSPF or internal BGP (iBGP) to clients that are outside of the ACI fabric. You can use route peering to configure OSPF or BGP peering on a service device so that the device can peer and exchange routes with the ACI leaf node to which it is connected.

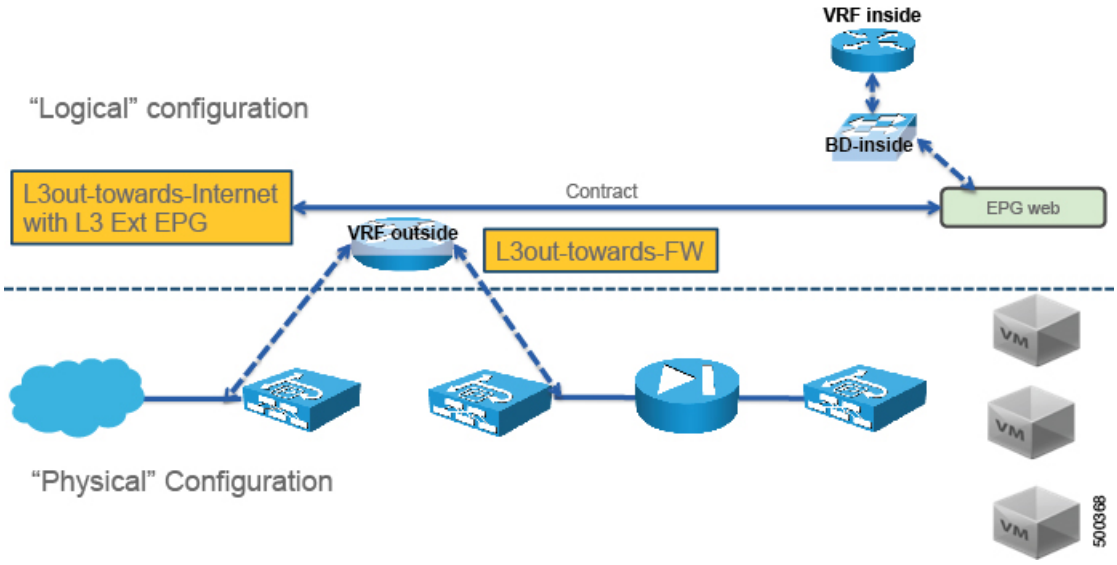
The goal for using route peering is to configure static routing to the firewall or load balancer and to use dynamic routing with the firewall or load balancer, as shown in the following figure:

***Figure 67: Route Peering***



Route peering requires 2 L3Outs, as shown in the following figure:

***Figure 68: The 2 L3Outs Required by Route Peering***




If you deploy route peering with a virtual appliance, you must specify the exact physical interface to which the virtual appliance is connected.

For more information about route peering, see *Cisco APIC Layer 4 to Layer 7 Services Deployment Guide* .

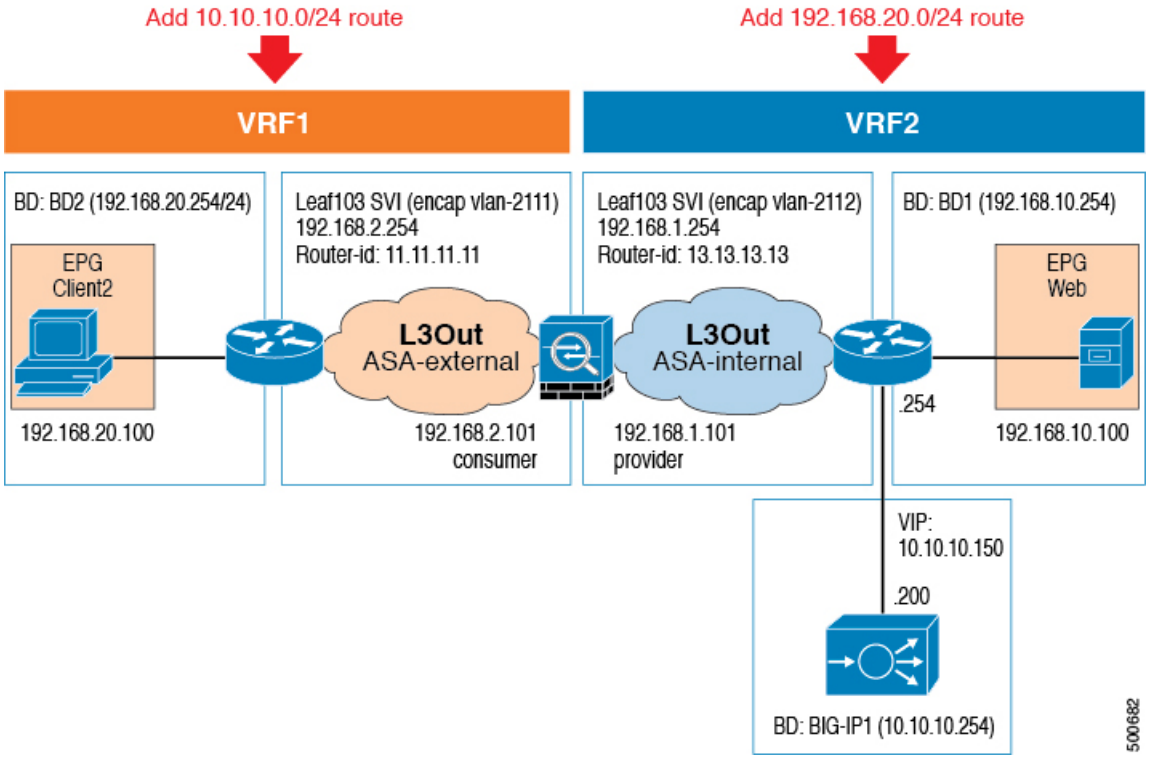
# Configuring Route Peering Using the GUI

The following procedure provides an example on how to configure route peering using an ASA device that is part of a two-node service graph. The other service device in the service graph is an F5 BIG-IP device.

This example provides values for most of the fields; the values for your setup will vary. You must fill out mandatory fields even if no example values are given in this procedure. This example uses `T1` as the name of the Tenant.

The following figure illustrates the components that you must configure to use route peering.

***Figure 69: Configurng Route Peering***




## Procedure

**Step 1** Create three bridge domains and two VRFs. This procedure uses `BD1`, `BD2`, and `BIG-IP1` as the bridge domains, and `VRF1` and `VRF2` as the VRFs.
See Creating Bridge Domains and VRFs Using the GUI, on page 36.

1. For `BD1`, in the **VRF** drop-down list, choose **Create VRF** to create `VRF2`.
2. For `BD2`, in the **VRF** drop-down list, choose **Create VRF** to create `VRF1`.
3. For `BIG-IP1`, in the **VRF** drop-down list, choose **VRF2**.

**Step 2** On the menu bar, choose **Tenants** > **All Tenants**.

**Step 3** In the Work pane, double click the tenant's name.

**Step 4** In the Navigation pane, choose **Tenant** ***tenant_name*** > **Security Policies** > **Contracts** > ***contract_name***. Choose the contract that you will associate with `VRF2`.

**Step 5** In the Work pane, choose the **Policy** tab.

**Step 6** In the **Scope** drop-down list, if the provider endpoint group and consumer endpoint group are in different tenants, choose **Global**. Otherwise, choose **Tenant**.

**Step 7** Click **Submit**.

**Step 8** Create an L3Out domain for ASA. On the menu bar, choose **Fabric** > **Access Policies**.

**Step 9** In the Navigation pane, choose **Physical and External Domains** > **External Routed Domains**.

**Step 10** In the Work pane, choose **Actions** > **Create Layer 3 Domain**.

**Step 11** In the **Create Layer 3 Domain** dialog box, fill in the fields as required, except as specified below:

a) In the **Name** field, enter `L3_ASA`.

b) In the **VLAN Pool** drop-down list, choose **Create VLAN Pool**.

**Step 12** In the **Create VLAN Pool** dialog box, fill in the fields as required, except as specified below:

a) In the **Name** field, enter `L3out-L4L7`.

b) In the **Encap Blocks** section, add a block with a **VLAN Range** of `2101-2199` and an **Allocation Mode** of **Static Allocation**.

**Step 13** Click **Submit**.

**Step 14** In the **Create Layer 3 Domain** dialog box, click **Submit**.

**Step 15** In the Work pane, verify that `L3_ASA` was created.

**Step 16** Create an external routed network with either a static route or OSPF.
To create an external routed network with a static route, see Configuring an External Routed Network for Route Peering with a Static Route Using the GUI, on page 115.

To create an external routed network with OSPF, see Configuring an External Routed Network for Route Peering with OSPF Using the GUI, on page 117.

In either case, use the values for the first external routed network.

**Step 17** Create a second external routed network using the same protocol as the previous step.
To create an external routed network with a static route, see Configuring an External Routed Network for Route Peering with a Static Route Using the GUI, on page 115.

To create an external routed network with OSPF, see Configuring an External Routed Network for Route Peering with OSPF Using the GUI, on page 117.

In either case, use the values for the second external routed network.

**Step 18** Create an ASA function profile in the `Common` tenant.
See Creating a Function Profile Using the GUI, on page 43.

The following differences in the steps are specific to route peering.

In the **Create Routed Outside** dialog box:

1. In the **Name** field, enter `ASA-routed`.
2. In the **Profile Group** drop-down list, choose **ASA-FP**.
3. In the **Copy Existing Profile Parameters** check box, put a check in the box.
4. In the **Profile** drop-down list, choose **CISCO-ASA-1.2/WebPolicyForRoutedMode**.
5. In the **Basic Parameters** section, configure the parameters as necessary. In the example setup, set the following parameters:

| L4-L7 Parameter or Folder | Usage and Notes |
|---|---|
| Device Config > Interface Related Configuration - externalIf > Interface Specific Configuration - externalIfCfg > IPv4 Address Configuration > IPv4 Address parameter. | Set the value to 192.168.2.101/255.255.255.0. |
| Device Config > Interface Related Configuration - externalIf > Static Routes List > IPv4 Route > Gateway parameter | Set the value to 192.168.2.254. |
| Device Config > Interface Related Configuration - externalIf > Static Routes List > IPv4 Route > Netmask parameter | Set the value to 255.255.255.0. |
| Device Config > Interface Related Configuration - externalIf > Static Routes List > IPv4 Route > Network parameter | Set the value to 192.168.20.0. |
| Device Config > Interface Related Configuration - internalIf > Interface Specific Configuration - internalIfCfg > IPv4 Address Configuration > IPv4 Address parameter | Set the value to 192.168.1.101/255.255.255.0. |
| Device Config > Interface Related Configuration - internalIf > Static Routes List > IPv4 Route > Gateway folder | Set the value to 192.168.1.254. |
| Device Config > Interface Related Configuration - internalIf > Static Routes List > IPv4 Route > Netmask parameter | Set the value to 255.255.255.0. |
| Device Config > Interface Related Configuration - internalIf > Static Routes List > IPv4 Route > Network | Set the value to 10.10.10.0. |

6 Click **Submit**.

**Step 19** Create a BIG-IP function profile in the Common tenant.
See Creating a Function Profile Using the GUI, on page 43.

The following differences in the steps are specific to this scenario.

1. In the **Name** field, enter `BIGIP-routed`.
2. In the **Profile Group** drop-down list, choose **BIGIP-FP**.
3. In the **Copy Existing Profile Parameters** check box, put a check in the box.
4. In the **Profile** drop-down list, choose **CISCO-BIGIP-1.2/WebPolicyForRoutedMode**.
5. In the **Basic Parameters** section, configure the parameters as necessary. In the example setup, set the following parameters:

| L4-L7 Parameter or Folder | Usage and Notes |
|---|---|
| `Device Config > LocalTraffic` folder | Define as `LocalTraffic-HTTP`. |
| `Device Config > LocalTraffic > Monitor` folder | Define as `Monitor`. |
| `Device Config > LocalTraffic > Monitor > Number of Monitor Failures to Trigger Service Down` parameter | Define as `FailByAttempts` with a value of `3`. |
| `Device Config > LocalTraffic > Monitor > Monitor Frequency` parameter | Define as `FrequencySeconds` with a value of `3`. |
| `Device Config > LocalTraffic > Monitor > Monitor Protocol` parameter | Define as `Type` with a value of `TCP`. |
| `Device Config > LocalTraffic > Pool` folder | Define as `Pool`. |
| `Device Config > LocalTraffic > Pool > Load Balancing Method` parameter | Define as `LBMethod` with a value of `ROUND_ROBIN`. |
| `Device Config > LocalTraffic > Pool > Pool Type` parameter | Define as `PoolType` with a value of `DYNAMIC`. |
| `Device Config > LocalTraffic > Pool > Pool Monitor` folder | Define as `PoolMonitor`. |
| `Device Config > LocalTraffic > Pool > Pool Monitor > Select Pool Monitor` parameter | Define as `PoolMonitorRel` with a value of `LocalTraffic-HTTP/Monitor`. |
| `Device Config > Network` folder | Define as `Network`. |
| `Device Config > Network > InternalSelfIP` folder | Define as `InternalSelfIP`. |
| `Device Config > Network > InternalSelfIP > Internal Self IP Address` parameter | Define as `SelfIPAddress` with a value of `10.10.10.200`. |

| L4-L7 Parameter or Folder | Usage and Notes |
|---|---|
| Device Config > Network > InternalSelfIP > Internal Self IP Netmask parameter | Define as SelfIPNetmask with a value of 255.255.255.0. |
| Device Config > Network > InternalSelfIP > Port Lockdown parameter | Define as PortLockdown with a value of NONE. |
| Device Config > Network > Route folder | Define as Route. |
| Device Config > Network > Route > Destination IP Address parameter | Define as DestinationIPAddress with a value of 0.0.0.0. |
| Device Config > Network > Route > Destination Netmask parameter | Define as DestinationNetmask with a value of 0.0.0.0. |
| Device Config > Network > Route > Next Hop Router IP Address parameter | Define as NextHopIPAddress with a value of 10.10.10.254. |
| Function Config > Listener folder | Define as Listener-HTTP. |
| Function Config > Listener > Protocol parameter | Define as Protocol with a value of TCP. |
| Function Config > Listener > Virtual Server IP Address parameter | Define as DestinationIPAddress with a value of 10.10.10.150. |
| Function Config > Listener > Virtual Server Netmask parameter | Define as DestinationIPAddress with a value of 255.255.255.255. |
| Function Config > Listener > Virtual Server Port parameter | Define as DestinationPort with a value of 80. |
| Function Config > Pool folder | Define as Pool. |
| Function Config > Pool > EPG Destination Port parameter | Define as EPGDestinationPort with a value of 80. |
| Function Config > Pool > Select Pool parameter | Define as PoolRel with a value of LocalTraffic-HTTP/Pool. |

6 Click **Submit**.

**Step 20** Create a service graph template.
See Creating a Layer 4 to Layer 7 Service Graph Template Using the GUI, on page 44.

The following differences in the steps are specific to route peering.

In the **Create L4-L7 Service Graph Template** dialog box:

1. In the **Graph Name** field, enter `FW-ADC-Graph-Peering`.
2. Drag the ASA device from the **Device Clusters** section and drop it between the consumer endpoint group and provider endpoint group to create a service node.
3. Drag the BIGIP device from the **Device Clusters** section and drop it between the consumer endpoint group and provider endpoint group, next to the ASA device, to create a service node.
4. In the **ASA-5525X-L3 Information** section, for the **Firewall** radio buttons choose **Routed** and for the **Profile** drop-down list choose **common/ASA-FP/ASA-routed**.
5. In the **BIGIP-LTM Information** section, for the **ADC** radio buttons choose **One-Arm** and for the **Profile** drop-down list choose **common/BIGIP-FP/BIGIP-oneARM-FP**.
6. Click **Submit**.

**Step 21** (Optional) In the Navigation pane, choose **Tenant** *tenant_name* > **L4-L7 Services** > **L4-L7 Service Graph Templates** > *template_name* > **Function Node -** *node_name* > **provider**.
Choose the service graph template that you just created and the load balancer function node.

a) In the **Attachment Notification** drop-down list, choose **Yes** if you want to use dynamic endpoint attach.

**Step 22** Apply the service graph template.
See Applying a Service Graph Template to Endpoint Groups Using the GUI, on page 45.

The following differences in the steps are specific to route peering.

In the **Apply L4-L7 Service Graph Template to EPGs** dialog box:

1. In the **Consumer EPG / External Network** drop-down list, choose **T1/test/epg-client2**.
2. In the **Provider EPG / External Network** drop-down list, choose **T1/test/epg-web**.
3. In the **Contract Information** section, fill out the fields as required.
4. Click **Next**.
5. In the **BIGIP-LTM Information** section, in the **BD** drop-down list, choose `T1/BIG-IP1`, and in the **Cluster Interface** drop-down list, choose `provider`.
6. In the **ASA-5525X-L3 Information** section, if you choose **Create Router Configuration**, then in the **Create Router Configuration** dialog box, in the **Name** field enter `ASA-RouterID`, and in the **Router ID** field enter `10.10.10.1`.
7. Click **Submit**.
8. In the **Apply L4-L7 Service Graph Template to EPGs** dialog box, in the **ASA-5525X-L3 Information** section, in the **Router Config** drop-down list, choose **T1/ASA-RouterID**.
9. In the **Consumer Connector** section, for the **Type** radio buttons, choose **Route Peering**.
10. In the **L3 Ext Network** drop-down list, choose **T1/ASA-external/ASA-external**.
11. In the **Cluster Interface** drop-down list, choose **consumer**.
12. In the **Provider Connector** section, for the **Type** radio buttons, choose **Route Peering**.
13. In the **L3 Ext Network** drop-down list, choose **T1/ASA-internal/ASA-internal**.
14. In the **Cluster Interface** drop-down list, choose **provider**.
15. Click **Next**.

16 Modify the parameter values if needed.

17 Click **Finish**.

# Configuring an External Routed Network for Route Peering with a Static Route Using the GUI

You can configure an external routed network for use with route peering by using a static route. The external routed network specifies the routing configuration in the Cisco Application Centric Infrastructure (ACI) fabric.

You must configure two external routed networks, and as such the following procedure provides two different sets of values—one for each of the networks—where necessary.

### Procedure

**Step 1** On the menu bar, choose **Tenants** > **All Tenants**.

**Step 2** In the Work pane, double click the tenant's name.

**Step 3** In the Navigation pane, choose ***tenant_name*** > **Networking** > **External Routed Networks**.

**Step 4** In the Work pane, choose **Actions** > **Create Routed Outside**.

**Step 5** In the **Create Routed Outside** dialog box, fill in the fields as required, except as specified below:

a) In the **Name** field, enter `ASA-external` for the first external routed network, or `ASA-internal` for the second external routed network.

b) In the **VRF** drop-down list, choose **T1/VRF1** for the first external routed network, or **T1/VRF2** for the second external routed network.

c) Do not put a check in either the **BGP** or **OSPF** check box.

d) In the **External Routed Domain** drop-down list, choose **L3_ASA**.

e) In the **Nodes and Interfaces Protocol Profiles** section, click +.

**Step 6** In the **Create Node Profile** dialog box, fill in the fields as required, except as specified below:

a) In the **Name** field, enter `Leaf3-NP`.

b) In the **Nodes** section, click +.

**Step 7** In the **Select Node** dialog box, fill in the fields as required, except as specified below:

a) In the **Node ID** drop-down list, choose **topology/pod-1/node-103**.

b) In the **Router ID** field, enter `11.11.11.11` for the first external routed network, or `13.13.13.13` for the second external routed network.

c) In the **Static Routes** section click +.

**Step 8** In the **Create Static Route** dialog box, fill in the fields as required, except as specified below:

a) In the **IP Address** field, enter `10.10.10.0/24` for the first external routed network, or `192.168.20.0/24` for the second external routed network.

b) In the **Prefix** section, enter a prefix for the static route.

c) In the **Next Hop Addresses** section, click +.

d) In the **Next Hop IP** column, enter `192.168.2.101` for the first external routed network, or `192.168.1.101` for the second external routed network.
e) Click **Update**.

**Step 9** Click **OK**.

**Step 10** In the **Select Node** dialog box, click **OK**.

**Step 11** In the **Interface Profiles** section, click +.

**Step 12** In the **Create Interface Profile** dialog box, fill in the fields as required, except as specified below:
a) In the **Name** field, enter `Leaf3-IP`.
b) In the **Interface** section, choose the **SVI** tab.

**Step 13** In the **Interface** section, click +.

**Step 14** In the **Select SVI Interface** dialog box, fill in the fields as required, except as specified below:
a) For the **Path Type** buttons, click **Direct Port Channel**.
b) In the **Path** drop-down list, choose **topology/pod-1/paths-103/pathep-[1G-PC-ASA]**.
c) In the **Encap** field, enter `vlan-2111` for the first external routed network, or `vlan-2112` for the second external routed network.
d) In the **IPv4 Primary / IPv6 Preferred Address** field, enter `192.168.2.254` for the first external routed network, or `192.168.1.254` for the second external routed network.
e) (Optional) In the **MTU (bytes)** field, change the value if necessary. This is the maximum transmission unit size, in bytes.
The default value is "inherit", which uses a default value of "9000" on the ACI and typically a default value of "1500" on the remote device. Having different MTU values can cause issues when peering between the ACI and the remote device. If the remote device's MTU value is set to "1500", then set the MTU value on the remote device's `L3Out` object to "9000" to match the ACI's MTU value.

**Step 15** Click **OK**.

**Step 16** In the **Create Interface Profile** dialog box, click **OK**.

**Step 17** In the **Create Node Profile** dialog box, click **OK**.

**Step 18** In the **Create Routed Outside** dialog box, click **Next**.

**Step 19** In the **External EPG Networks** section, click +.

**Step 20** In the **Create External Network** dialog box, fill in the fields as required, except as specified below:
a) In the **Name** field, enter `ASA-external` for the first external routed network, or `ASA-internal` for the second external routed network.
b) In the **Subnet** section, click +.

**Step 21** In the **Create Subnet** dialog box, fill in the fields as required, except as specified below:
a) In the **IP Address** field, enter `10.10.10.0/24` for the first external routed network, or `192.168.20.0/24` for the second external routed network.
b) In the **Scope** section, put a check in the **External Subnets for the External EPG** check box.

**Step 22** Click **OK**.

**Step 23** (Optional) Create additional subnets as needed.

**Step 24** In the **Create External Network** dialog box, click **OK**.

**Step 25** In the **Create Routed Outside** dialog box, click **Finish**.

# Configuring an External Routed Network for Route Peering with OSPF Using the GUI

You can configure an external routed network for use with route peering by using OSPF. The external routed network specifies the routing configuration in the Cisco Application Centric Infrastructure (ACI) fabric.

You must configure two external routed networks, and as such the following procedure provides two different sets of values—one for each of the networks—where necessary.

### Procedure

**Step 1** On the menu bar, choose **Tenants** > **All Tenants**.

**Step 2** In the Work pane, double click the tenant's name.

**Step 3** In the Navigation pane, choose ***tenant_name*** > **Networking** > **External Routed Networks**.

**Step 4** In the Work pane, choose **Actions** > **Create Routed Outside**.

**Step 5** In the **Create Routed Outside** dialog box, fill in the fields as required, except as specified below:

a) In the **Name** field, enter `ASA-external` for the first external routed network, or `ASA-internal` for the second external routed network.

b) In the **VRF** drop-down list, choose **T1/VRF1** for the first external routed network, or **T1/VRF2** for the second external routed network.

c) Put a check in the **OSPF** check box.

d) In the **OSPF Area ID** field, enter `0.0.0.1`.

e) For the **OSPF Area Type** buttons, click **Regular area**.

f) In the **External Routed Domain** drop-down list, choose **L3_ASA**.

g) In the **Nodes and Interfaces Protocol Profiles** section, click +.

**Step 6** In the **Create Node Profile** dialog box, fill in the fields as required, except as specified below:

a) In the **Name** field, enter `Leaf3-NP`.

b) In the **Nodes** section, click +.

**Step 7** In the **Select Node** dialog box, fill in the fields as required, except as specified below:

a) In the **Node ID** drop-down list, choose **topology/pod-1/node-103**.

b) In the **Router ID** field, enter `11.11.11.11` for the first external routed network, or `13.13.13.13` for the second external routed network.

c) In the **Static Routes** section click +.

**Step 8** Click **OK**.

**Step 9** In the **Select Node** dialog box, click **OK**.

**Step 10** In the **Interface Profiles** section, click +.

**Step 11** In the **Create Interface Profile** dialog box, fill in the fields as required, except as specified below:

a) In the **Name** field, enter `Leaf3-IP`.

b) In the **Interface** section, choose the **SVI** tab.

**Step 12** In the **Interface** section, click +.

**Step 13** In the **Select SVI Interface** dialog box, fill in the fields as required, except as specified below:

a) For the **Path Type** buttons, click **Direct Port Channel**.

b) In the **Path** drop-down list, choose **topology/pod-1/paths-103/pathep-[1G-PC-ASA]**.
c) In the **Encap** field, enter `vlan-2111` for the first external routed network, or `vlan-2112` for the second external routed network.
d) In the **IPv4 Primary / IPv6 Preferred Address** field, enter `192.168.2.254` for the first external routed network, or `192.168.1.254` for the second external routed network.
e) (Optional) In the **MTU (bytes)** field, change the value if necessary. This is the maximum transmission unit size, in bytes.
The default value is "inherit", which uses a default value of "9000" on the ACI and typically a default value of "1500" on the remote device. Having different MTU values can cause issues when peering between the ACI and the remote device. If the remote device's MTU value is set to "1500", then set the MTU value on the remote device's `L3Out` object to "9000" to match the ACI's MTU value.

**Step 14** Click **OK**.
**Step 15** In the **Create Interface Profile** dialog box, click **OK**.
**Step 16** In the **Create Node Profile** dialog box, click **OK**.
**Step 17** In the **Create Routed Outside** dialog box, click **Next**.
**Step 18** In the **External EPG Networks** section, click **+**.
**Step 19** In the **Create External Network** dialog box, fill in the fields as required, except as specified below:
a) In the **Name** field, enter `ASA-external` for the first external routed network, or `ASA-internal` for the second external routed network.
b) In the **Subnet** section, click **+**.
**Step 20** In the **Create Subnet** dialog box, fill in the fields as required, except as specified below:
a) In the **IP Address** field, enter `192.168.20.0/24` for the first external routed network (`ASA-external`), or `10.10.10.0/24` for the second external routed network (`ASA-internal`).
b) In the **Scope** field, choose **Export Route Control Subnet**.
**Step 21** Click **OK**.
**Step 22** (Optional) Create additional subnets as needed.
**Step 23** In the **Create External Network** dialog box, click **OK**.
**Step 24** In the **Create Routed Outside** dialog box, click **Finish**.
**Step 25** In the Navigation pane, choose ***tenant_name*** > **Networking** > **VRFs** > **VRF1** for the first external routed network, or ***tenant_name*** > **Networking** > **VRFs** > **VRF2** for the second external routed network..
**Step 26** In the Work pane, in the **Route Tag Policy** drop-down list, choose **Create Route Tag Policy**.
**Step 27** In the **Create Route Tag Policy** dialog box, fill in the fields as required, except as specified below:
a) In the **Name** field, enter `Tag-100` for the first external routed network, or `Tag-200` for the second external routed network.
b) In the **Tag** drop-down list, choose **100** for the first external routed network, or `200` for the second external routed network.
c) Click **Submit**.
**Step 28** Click **Submit**.

# Verifying a Route Peering With a Static Route Configuration Using the GUI

After configuring a setup to use route peering with a static route, you can verify the configuration with the following procedure.

### Procedure

**Step 1** Verify the service graph deployment.
See Verifying a Service Graph Deployment Using the GUI, on page 47

For the deployed devices, you should see `ASA-5525X-L3-none` and `BIGIP-LTM-VRF2`.

For the ASA-5525X-L3 cluster interfaces, you should see `ASA-5525X-L3_consumer` and `ASA-5525X-L3_provider`[

**Step 2** Using the CLI on the leaf switch, verify that the IP routing table for `VRF1` is correct.

```
Leaf3# show ip route vrf T1:VRF1
...
10.10.10.0/24, ubest/mbest: 1/0
    *via 192.168.2.101, vlan15, [1/0], 17:29:50, static
11.11.11.11/32, ubest/mbest: 2/0, attached, direct
    *via 11.11.11.11, lo3, [1/0], 4d23h, local, local
    *via 11.11.11.11, lo3, [1/0], 4d23h, direct
192.168.2.0/24, ubest/mbest: 1/0, attached, direct
    *via 192.168.2.254, vlan15, [1/0], 17:29:50, direct
192.168.2.254/32, ubest/mbest: 1/0, attached
    *via 192.168.2.254, vlan15, [1/0], 17:29:50, local, local
192.168.20.0/24, ubest/mbest: 1/0, attached, direct, pervasive
    *via 10.0.80.64%overlay-1, [1/0], 01:53:21, static
```

The route peering IP route is shown in bold.

**Step 3** Verify that the IP routing table for `VRF2` is correct.

```
Leaf3# show ip route vrf T1:VRF2
...
10.10.10.0/24, ubest/mbest: 1/0, attached, direct, pervasive
    *via 10.0.80.64%overlay-1, [1/0], 01:54:10, static
10.10.10.254/32, ubest/mbest: 1/0, attached
    *via 10.10.10.254, vlan17, [1/0], 01:54:10, local, local
192.168.1.0/24, ubest/mbest: 1/0, attached, direct
    *via 192.168.1.254, vlan16, [1/0], 02:08:12, direct
192.168.1.254/32, ubest/mbest: 1/0, attached
    *via 192.168.1.254, vlan16, [1/0], 02:08:12, local, local
192.168.10.0/24, ubest/mbest: 1/0, attached, direct, pervasive
    *via 10.0.80.64%overlay-1, [1/0], 01:54:10, static
192.168.20.0/24, ubest/mbest: 1/0
    *via 192.168.1.101, vlan16, [1/0], 02:08:12, static
```

The route peering IP route is shown in bold.

**Step 4** Verify that the routing table is correct.

```
ASA5525X/T1# show route
...
S*    0.0.0.0 0.0.0.0 [1/0] via 172.16.255.254, management
S        10.10.10.0 255.255.255.0 [1/0] via 192.168.1.254, internalIf
C        172.16.0.0 255.255.0.0 is directly connected, management
L        172.16.0.101 255.255.255.255 is directly connected, management
C        192.168.1.0 255.255.255.0 is directly connected, internalIf
L        192.168.1.101 255.255.255.255 is directly connected, internalIf
C        192.168.2.0 255.255.255.0 is directly connected, externalIf
L        192.168.2.101 255.255.255.255 is directly connected, externalIf
S     192.168.20.0 255.255.255.0 [1/0] via 192.168.2.254, externalIf
```

The route peering routes are shown in bold.

# Verifying a Route Peering With OSPF Configuration Using the GUI

After configuring a setup to use route peering with OSPF, you can verify the configuration with the following procedure.

### Procedure

**Step 1** Verify the service graph deployment.
See Verifying a Service Graph Deployment Using the GUI, on page 47

For the deployed devices, you should see `ASA-5525X-L3-none` and `BIGIP-LTM-VRF2`.

For the ASA-5525X-L3 cluster interfaces, you should see `ASA-5525X-L3_consumer` and `ASA-5525X-L3_provider`[

**Step 2** Using the CLI on the leaf switch, verify that the IP routing table for `VRF1` is correct.

```
Leaf3# show ip route vrf T1:VRF1
...
10.10.10.0/24, ubest/mbest: 1/0
    *via 192.168.2.101, vlan20, [110/20], 00:00:27, ospf-default, type-2, tag 200
11.11.11.11/32, ubest/mbest: 2/0, attached, direct
    *via 11.11.11.11, lo3, [1/0], 5d02h, local, local
    *via 11.11.11.11, lo3, [1/0], 5d02h, direct
192.168.1.0/24, ubest/mbest: 1/0
    *via 192.168.2.101, vlan20, [110/14], 00:15:51, ospf-default, intra
192.168.2.0/24, ubest/mbest: 1/0, attached, direct
    *via 192.168.2.254, vlan20, [1/0], 00:30:04, direct
192.168.2.254/32, ubest/mbest: 1/0, attached
    *via 192.168.2.254, vlan20, [1/0], 00:30:04, local, local
192.168.20.0/24, ubest/mbest: 1/0, attached, direct, pervasive
    *via 10.0.80.64%overlay-1, [1/0], 00:16:02, static
```

The route peering IP route is shown in bold.

**Step 3** Verify that the IP routing table for VRF2 is correct.

```
Leaf3# show ip route vrf T1:VRF2
...
10.10.10.0/24, ubest/mbest: 1/0, attached, direct, pervasive
    *via 10.0.80.64%overlay-1, [1/0], 00:16:05, static
10.10.10.254/32, ubest/mbest: 1/0, attached
    *via 10.10.10.254, vlan13, [1/0], 00:16:05, local, local
192.168.1.0/24, ubest/mbest: 1/0, attached, direct
    *via 192.168.1.254, vlan16, [1/0], 04:48:44, direct
192.168.1.254/32, ubest/mbest: 1/0, attached
    *via 192.168.1.254, vlan16, [1/0], 04:48:44, local, local
192.168.2.0/24, ubest/mbest: 1/0
    *via 192.168.1.101, vlan16, [110/14], 00:15:53, ospf-default, intra
192.168.10.0/24, ubest/mbest: 1/0, attached, direct, pervasive
    *via 10.0.80.64%overlay-1, [1/0], 00:01:52, static
192.168.20.0/24, ubest/mbest: 1/0
    *via 192.168.1.101, vlan16, [110/20], 00:01:48, ospf-default, type-2, tag 100
```

The route peering IP route is shown in bold.

**Step 4** Verify that the routing table is correct.

```
ASA5525X/T1# show route
...
S*    0.0.0.0 0.0.0.0 [1/0] via 172.16.255.254, management
O E2     10.10.10.0 255.255.255.0
           [110/20] via 192.168.1.254, 00:00:32, internalIf
C        172.16.0.0 255.255.0.0 is directly connected, management
L        172.16.0.101 255.255.255.255 is directly connected, management
C        192.168.1.0 255.255.255.0 is directly connected, internalIf
L        192.168.1.101 255.255.255.255 is directly connected, internalIf
C        192.168.2.0 255.255.255.0 is directly connected, externalIf
L        192.168.2.101 255.255.255.255 is directly connected, externalIf
O E2     192.168.20.0 255.255.255.0
           [110/20] via 192.168.2.254, 00:00:32, externalIf
```

The route peering routes are shown in bold.