Cisco APIC Basic Configuration Guide, Release 1.x

First Published: 2015-10-19

Last Modified: 2016-12-02

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
- Server 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

Cisco Application Centric Infrastructure (ACI) Documentation

The ACI documentation is 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 (ACI) Simulator Documentation

The 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 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.

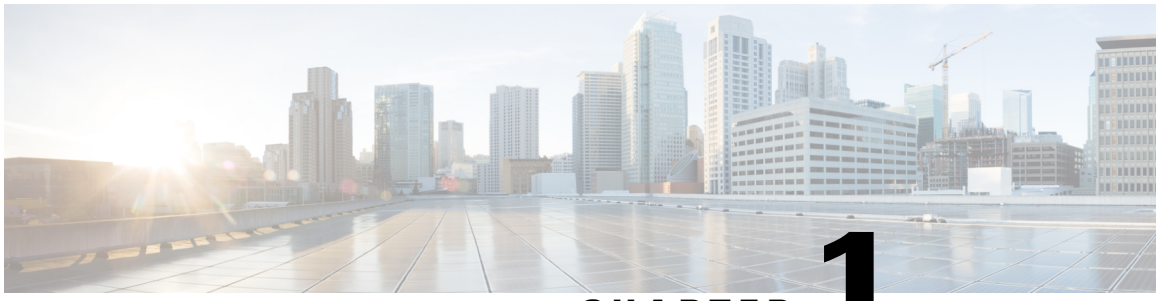
Documentation Feedback

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

User Access, Authentication, and Accounting

This chapter contains the following sections:

Access Rights Workflow Dependencies

The Cisco ACI RBAC rules enable or restrict access to some or all of the fabric. For example, in order to configure a leaf switch for bare metal server access, the logged in administrator must have rights to the `infra` domain. By default, a tenant administrator does not have rights to the `infra` domain. In this case, a tenant administrator who plans to use a bare metal server connected to a leaf switch could not complete all the necessary steps to do so. The tenant administrator would have to coordinate with a fabric administrator who has rights to the `infra` domain. The fabric administrator would set up the switch configuration policies that the tenant administrator would use to deploy an application policy that uses the bare metal server attached to an ACI leaf switch.

User Access, Authentication, and Accounting

APIC policies manage the access, authentication, and accounting (AAA) functions of the Cisco ACI fabric. The combination of user privileges, roles, and domains with access rights inheritance enables administrators to configure AAA functions at the managed object level in a very granular fashion. These configurations can be implemented using the REST API, the CLI, or the GUI.

Multiple Tenant Support

A core APIC internal data access control system provides multitenant isolation and prevents information privacy from being compromised across tenants. Read/write restrictions prevent any tenant from seeing any other tenant's configuration, statistics, faults, or event data. Unless the administrator assigns permissions to do so, tenants are restricted from reading fabric configuration, policies, statistics, faults, or events.

User Access: Roles, Privileges, and Security Domains

The APIC provides access according to a user's role through role-based access control (RBAC). An ACI fabric user is associated with the following:

- A set of roles
- For each role, a privilege type: no access, read-only, or read-write
- One or more security domain tags that identify the portions of the management information tree (MIT) that a user can access

The ACI fabric manages access privileges at the managed object (MO) level. A privilege is an MO that enables or restricts access to a particular function within the system. For example, fabric-equipment is a privilege bit. This bit is set by the APIC on all objects that correspond to equipment in the physical fabric.

A role is a collection of privilege bits. For example, because an "admin" role is configured with privilege bits for "fabric-equipment" and "tenant-security," the "admin" role has access to all objects that correspond to equipment of the fabric and tenant security.

A security domain is a tag associated with a certain subtree in the ACI MIT object hierarchy. For example, the default tenant "common" has a domain tag `common`. Similarly, the special domain tag `all` includes the entire MIT object tree. An administrator can assign custom domain tags to the MIT object hierarchy. For example, an administrator could assign the "solar" domain tag to the tenant named solar. Within the MIT, only certain objects can be tagged as security domains. For example, a tenant can be tagged as a security domain but objects within a tenant cannot.

Creating a user and assigning a role to that user does not enable access rights. It is necessary to also assign the user to one or more security domains. By default, the ACI fabric includes two special pre-created domains:

- `All`—allows access to the entire MIT
- `Infra`— allows access to fabric infrastructure objects/subtrees, such as fabric access policies

Note For read operations to the managed objects that a user's credentials do not allow, a "DN/Class Not Found" error is returned, not "DN/Class Unauthorized to read." For write operations to a managed object that a user's credentials do not allow, an HTTP 401 Unauthorized error is returned. In the GUI, actions that a user's credentials do not allow, either they are not presented, or they are greyed out.

A set of pre-defined managed object classes can be associated with domains. These classes should not have overlapping containment. Examples of classes that support domain association are as follows:

- Layer 2 and Layer 3 network managed objects
- Network profiles (such as physical, Layer 2, Layer 3, management)
- QoS policies

When an object that can be associated with a domain is created, the user must assign domain(s) to the object within the limits of the user's access rights. Domain assignment can be modified at any time.

If a virtual machine management (VMM) domain is tagged as a security domain, the users contained in the security domain can access the correspondingly tagged VMM domain. For example, if a tenant named solar is tagged with the security domain called sun and a VMM domain is also tagged with the security domain called sun, then users in the solar tenant can access the VMM domain according to their access rights.

Configuring a Local User

In the initial configuration script, the admin account is configured and the admin is the only user when the system starts. The APIC supports a granular, role-based access control system where user accounts can be created with various roles including non-admin users with fewer privileges.

Configuring a Local User Using the GUI

Note To watch an example video of this task, see Videos Webpage.

Before You Begin

- The ACI fabric is installed, APIC controllers are online, and the APIC cluster is formed and healthy.
- As appropriate, the security domain(s) that the user will access are defined. For example, if the new use account will be restricted to accessing a tenant, the tenant domain is tagged accordingly.
- An APIC user account is available that will enable the following:
  - Creating the TACACS+ and TACACS+ provider group.
  - Creating the local user account in the target security domain(s). If the target domain is `all`, the login account used to create the new local user must be a fabric-wide administrator that has access to `all`. If the target domain is a tenant, the login account used to create the new local user must be a tenant administrator that has full read write access rights to the target tenant domain.

Procedure

Step 1 On the menu bar, choose **ADMIN** > **AAA**.

Step 2 In the **Navigation** pane, click **AAA Authentication**.

Step 3 In the **Work** pane, verify that in the default **Authentication** field, the **Realm** field displays as Local.

Step 4 In the **Navigation** pane, expand **Security Management** > **Local Users**.
The admin user is present by default.

Step 5 In the **Navigation** pane, right-click **Create Local User**.

Step 6 In the **Security** dialog box, choose the desired security domain for the user, and click **Next**.

Step 7 In the **Roles** dialog box, click the radio buttons to choose the roles for your user, and click **Next**.
You can provide read-only or read/write privileges.

Step 8 In the **User Identity** dialog box, perform the following actions:

a) In the **Login ID** field, add an ID.

b) In the **Password** field, enter the password.
At the time a user sets their password, the APIC validates it against the following criteria:

- Minimum password length is 8 characters.
- Maximum password length is 64 characters.
- Has fewer than three consecutive repeated characters.
- Must have characters from at least three of the following characters types: lowercase, uppercase, digit, symbol.
- Does not use easily guessed passwords.
- Cannot be the username or the reverse of the username.
- Cannot be any variation of cisco, isco or any permutation of these characters or variants obtained by changing the capitalization of letters therein.

c) In the **Confirm Password** field, confirm the password.

d) Click **Finish**.

Step 9 In the **Navigation** pane, click the name of the user that you created. In the **Work** pane, expand the + sign next to your user in the **Security Domains** area.
The access privileges for your user are displayed.

Configuring a Local User Using the NX-OS Style CLI

In the initial configuration script, the admin account is configured and the admin is the only user when the system starts. The APIC supports a granular, role-based access control system where user accounts can be created with various roles including non-admin users with fewer privileges.

Configuring a Local User Using the NX-OS Style CLI

Procedure

Step 1 In the NX-OS CLI, start in configuration mode, shown as follows:

Example:

```
apic1# configure
apic1(config)#
```

Step 2 Create a new user, shown as follows:

Example:

```
apic1(config)# username
 WORD       User name (Max Size 28)
 admin
 cli-user
 jigarshah
 test1
 testUser

apic1(config)# username test
apic1(config-username)#
 account-status      Set The status of the locally-authenticated user account.
 certificate         Create AAA user certificate in X.509 format.
 clear-pwd-history   Clears the password history of a locally-authenticated user
 domain              Create the AAA domain to which the user belongs.
 email               Set The email address of the locally-authenticated user.
 exit                Exit from current mode
 expiration          If expires enabled, Set expiration date of locally-authenticated user
 account.
 expires             Enable expiry for locally-authenticated user account
 fabric              show fabric related information
 first-name          Set the first name of the locally-authenticated user.
 last-name           Set The last name of the locally-authenticated user.
 no                  Negate a command or set its defaults
 password            Set The system user password.
 phone               Set The phone number of the locally-authenticated user.
 pwd-lifetime        Set The lifetime of the locally-authenticated user password.
 pwd-strength-check  Enforces the strength of the user password
 show                Show running system information
 ssh-key             Update ssh key for the user for ssh authentication
 where               show the current mode

apic1(config-username)# exit
```

Configuring a Remote User

Instead of configuring local users, you can point the APIC at the centralized enterprise credential datacenter. The APIC supports Lightweight Directory Access Protocol (LDAP), active directory, RADIUS, and TACACS+.

To configure a remote user authenticated through an external authentication provider, you must meet the following prerequisites:

- The DNS configuration should have already been resolved with the hostname of the RADIUS server.

- You must configure the management subnet.

AV Pair on the External Authentication Server

You can add a Cisco attribute/value (AV) pair to the existing user record to propagate the user privileges to the APIC controller. The Cisco AV pair is a single string that you use to specify the Role-Based Access Control (RBAC) roles and privileges for an APIC user. An example configuration for an open RADIUS server (/etc/raddb/users) is as follows:

```
aaa-network-admin Cleartext-Password := "<password>"
Cisco-avpair = "shell:domains = all/aaa/read-all(16001)"
```

Best Practice for Assigning AV Pairs

As best practice, Cisco recommends that you assign unique UNIX user ids in the range 16000-23999 for the AV Pairs that are assigned to users when in bash shell (using SSH, Telnet or Serial/KVM consoles). If a situation arises when the Cisco AV Pair does not provide a UNIX user id, the user is assigned a user id of 23999 or similar number from the range that also enables the user's home directories, files, and processes accessible to remote users with a UNIX ID of 23999.

Configuring an AV Pair on the External Authentication Server

The numerical value within the parentheses in the attribute/value (AV) pair string is used as the UNIX user ID of the user who is logged in using Secure Shell (SSH) or Telnet.

Procedure

Configure an AV pair on the external authentication server.
The Cisco AV pair definition is as follows (Cisco supports AV pairs with and without UNIX user IDs specified):

Example:

```
* shell:domains =
domainA/writeRole1|writeRole2|writeRole3/readRole1|readRole2,domainB/writeRole1|writeRole2|writeRole3/readRole1|readRole2

         * shell:domains =
domainA/writeRole1|writeRole2|writeRole3/readRole1|readRole2,domainB/writeRole1|writeRole2|writeRole3/readRole1|readRole2(8101)

These are the boost regexes supported by APIC:
uid_regex("shell:domains\\s*[=:]\\s*((\\S+?/\\S*?/\\S*?)(,\\S+?/\\S*?/\\S*?){0,31})(\\(\\d+\\))$");
regex("shell:domains\\s*[=:]\\s*((\\S+?/\\S*?/\\S*?)(,\\S+?/\\S*?/\\S*?){0,31})$");
```

The following is an example:

```
shell:domains = coke/tenant-admin/read-all,pepsi//read-all(16001)
```

Configuring APIC for TACACS+ Access

Before You Begin

- The ACI fabric is installed, APIC controllers are online, and the APIC cluster is formed and healthy.
- The TACACS+ server host name or IP address, port, and key are available.
- The APIC management EPG is available.

- An APIC user account is available that will enable the following:
  - Creating the TACACS+ provider and TACACS+ provider group.

Figure 1: TACACS+ Provider

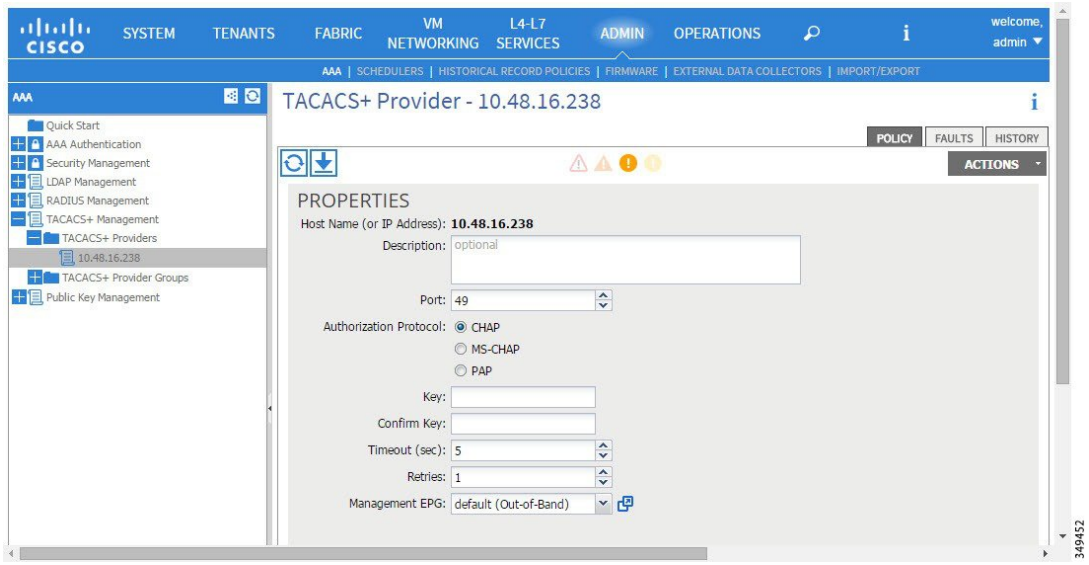
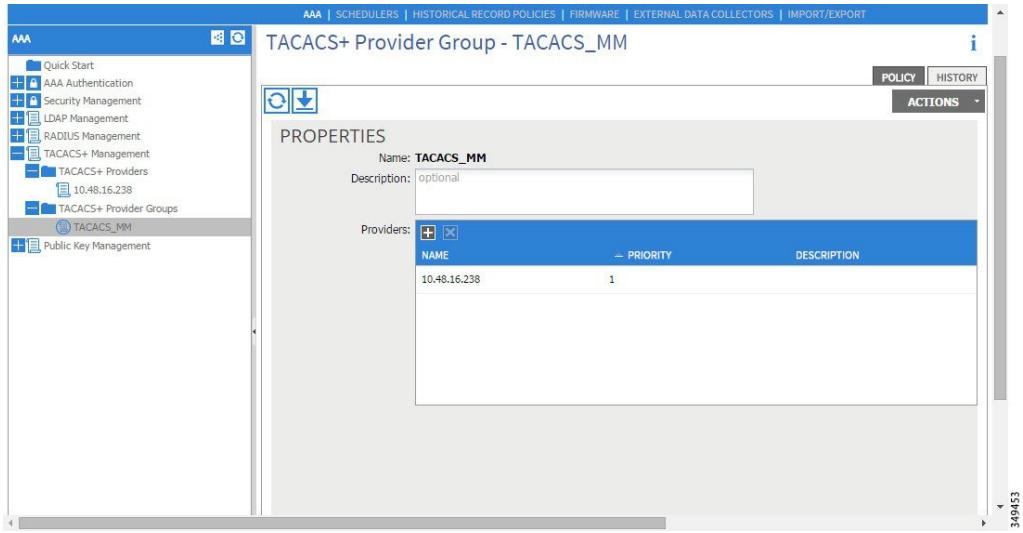
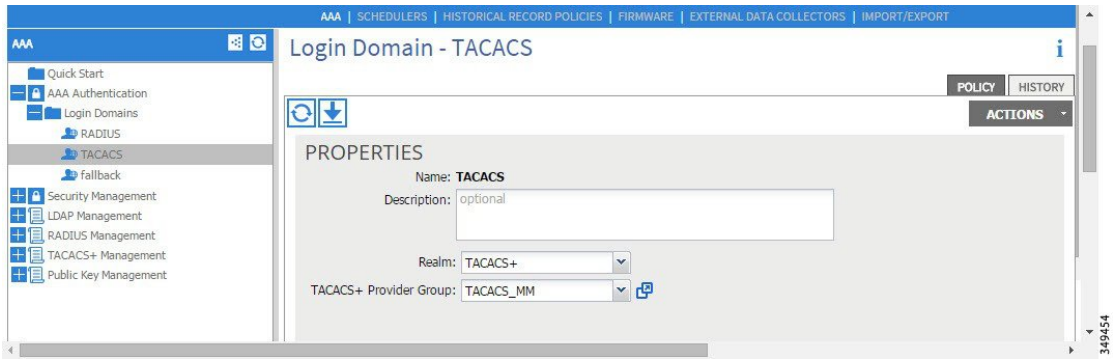


Figure 2: TACACS+ Provider Group



- Creating the TACACS+ login domain.

Figure 3: AAA Login Domain for TACACS+



Procedure

Step 1 On the APIC, create the **TACACS+ Provider**.

a) On the APIC menu bar, click **ADMIN** > **AAA**.

b) In the **Navigation** pane, click the + icon to expand the**TACACS+ Managment** option.

c) In the **Navigation** pane, right-click the **TACACS+ Providers** option, and select **Create TACACS+ Provider**.

d) Specify the TACACS+ host name (or IP address), port, authorization protocol, key, and management EPG.

Note If the APIC is configured for in-band management connectivity, choosing an out-of-band management EPG for TACACS+ access does not take effect. Alternatively, an out-of-band over an in-band management EPG can connect a TACACS+ server but requires configuring a static route for the TACACS+ server. The Cisco ACS sample configuration procedure below uses an APIC in-band IP address.

Step 2 Create the **TACACS+ Provider Group**.

a) In the **Navigation** pane, right-click the the**TACACS+ Provider Groups** option, and select **Create TACACS+ Provider Group**

b) Specify the TACACS+ Provider Group name, description, and provider(s) as appropriate.

Step 3 Create the **Login Domain** for TACACS+.

a) In the **Navigation** pane, click the + icon to expand the**AAA Authentication** option.

b) In the **Navigation** pane, right-click the **Login Domains** option, and select **Create Login Domain**.

c) Specify the Login Domain name, description, realm, and provider group as appropriate.

What to Do Next

This completes the APIC TACACS+ configuration steps. Next, if a RAIDUS server will also be used, configure the APIC for RADIUS. If only a TACACS+ server will be used, go to the ACS server configuration topic below.

Configuring APIC for RADIUS Access

Before You Begin

- The ACI fabric is installed, APIC controllers are online, and the APIC cluster is formed and healthy.
- The RADIUS server host name or IP address, port, authorization protocol, and key are available.
- The APIC management EPG is available.
- An APIC user account is available that will enable the following:

- Creating the RADIUS provider and RADIUS provider group.

Figure 4: RADIUS Provider

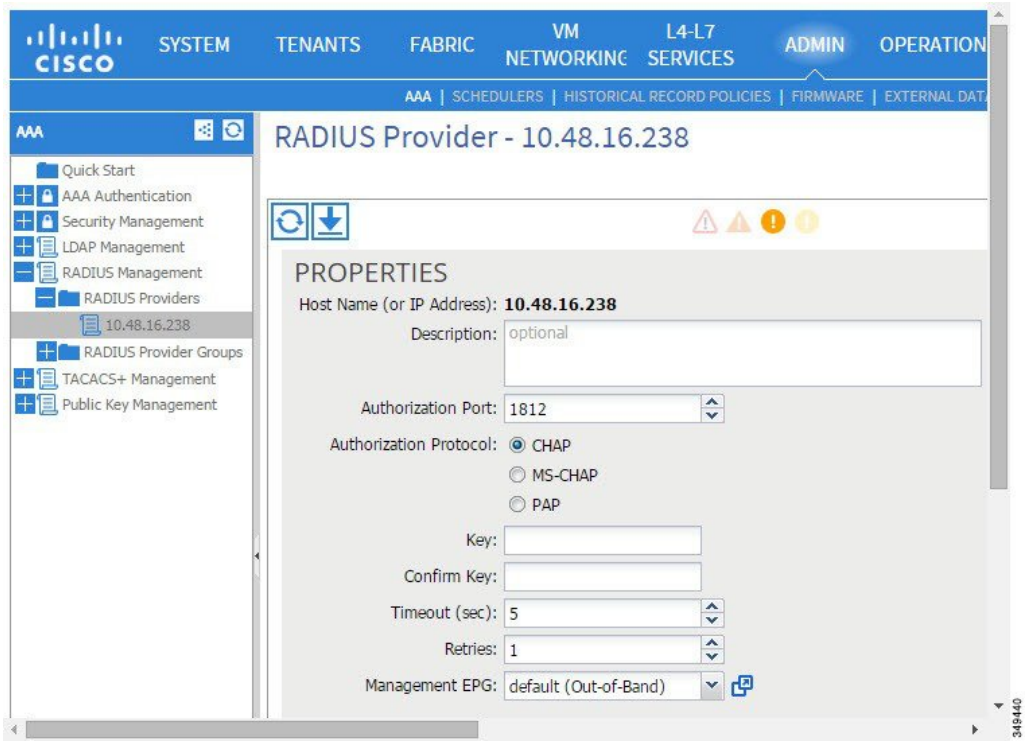
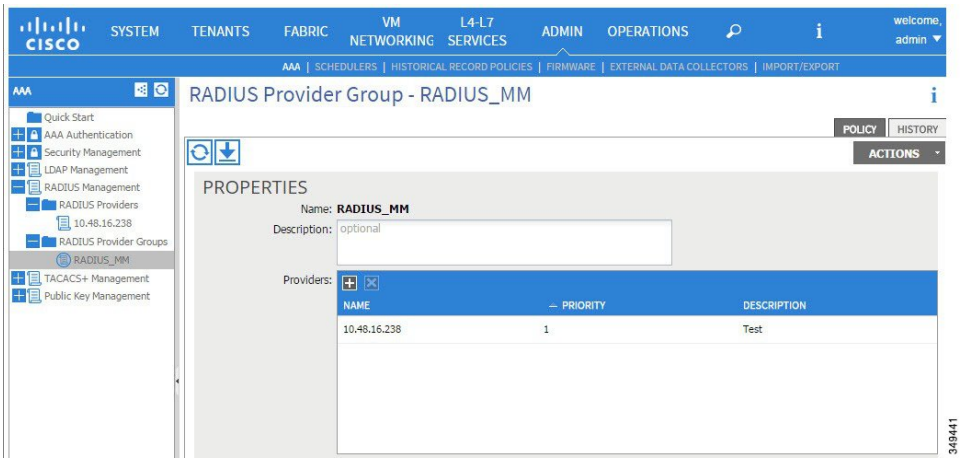
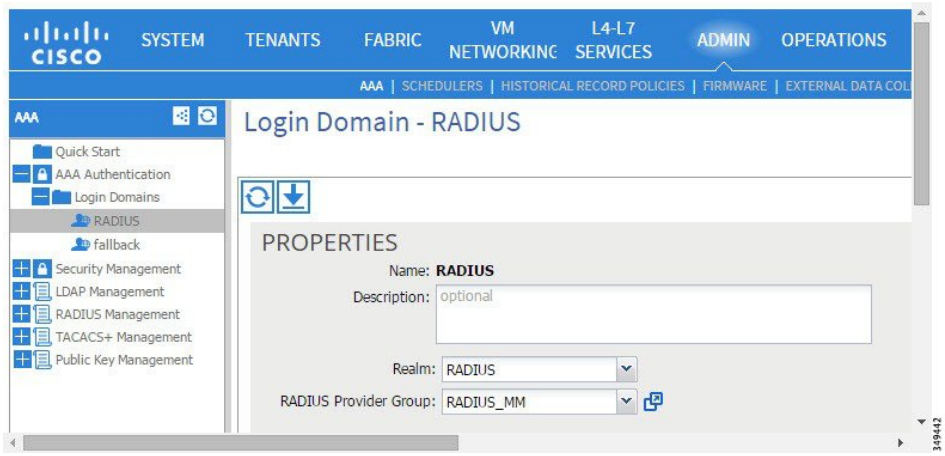


Figure 5: RADIUS Provider Group



- Creating the RADIUS login domain.

Figure 6: AAA Login Domain for RADIUS



Procedure

Step 1 On the APIC, create the **RADIUS Provider**.

a) On the APIC menu bar, click **ADMIN** > **AAA**.

b) In the **Navigation** pane, click the + icon to expand the**RADIUS Managment** option.

c) In the **Navigation** pane, right-click the **RADIUS Providers** option, and select **Create RADIUS Provider**.

d) Specify the RADIUS host name (or IP address), port, protocol, and management EPG.

Note If the APIC is configured for in-band management connectivity, choosing an out-of-band management EPG for RADIUS access does not take effect. Alternatively, an out-of-band over an in-band management EPG can connect a RADIUS server but requires configuring a static route for the RADIUS server. The Cisco ACS sample configuration procedure below uses an APIC in-band IP address.

Step 2 Create the **RADIUS Provider Group**.

a) In the **Navigation** pane, right-click the the**RADIUS Provider Groups** option, and select **Create RADIUS Provider Group**

b) Specify the RADIUS Provider Group name, description, and provider(s) as appropriate.

Step 3 Create the **Login Domain** for RADIUS.

a) In the **Navigation** pane, click the + icon to expand the**AAA Authentication** option.

b) In the **Navigation** pane, right-click the **Login Domains** option, and select **Create Login Domain**.

c) Specify the Login Domain name, description, realm, and provider group as appropriate.

What to Do Next

This completes the APIC RADIUS configuration steps. Next, configure the RADIUS server.

Configuring A Cisco Secure Access Control Server for RADIUS and TACACS+ Access to the APIC

Before You Begin

- The Cisco Secure Access Control Server (ACS) version 5.5 is installed and online.

Note ACS v5.5 was used to document these steps. Other versions of ACS might support this task but the GUI procedures might vary accordingly.

- The APIC RADIUS or TACACS+ keys are available (or keys for both if both will be configured).
- The APIC controllers are installed and online; the APIC cluster is formed and healthy.
- The RADIUS or TACACS+ port, authorization protocol, and key are available.

Procedure

Step 1 Log in to the ACS server to configure the APIC as a client.

a) Navigate to **Network Resources** > **Network Devices Groups** > **Network Devices and AAA Clients**.

b) Specify the client name, the APIC in-band IP address, select the TACACS+ or RADIUS (or both) authentication options.

Note If the only RADIUS or TACACS+ authentication is needed, select only the needed option.

c) Specify the authentication details such as Shared Secret (key), and port as appropriate for the authentication option(s).

Note The **Shared Secret**(s) must match the APIC **Provider** key(s).

Step 2 Create the Identity Group.

a) Navigate to **Users and Identity Stores** > **Internal Groups** option.

b) Specify the **Name**, and **Parent Group** as appropriate.

Step 3 Map users to the Identity Group.

a) In the **Navigation** pane, click the **Users and Identity Stores** > **Internal Identity Stores** > **Users** option.

b) Specify the user **Name**, and **Identity Group** as appropriate.

Step 4 Create the Policy Element.

a) Navigate to the **Policy Elements** option.

b) For RADIUS, specify the Authorization and Permissions > Network Access > Authorization Profiles **Name**. For TACACS+, specify the Authorization and Permissions > Device Administration > Shell Profile **Name**as appropriate.

c) For RADIUS, specify the **Attribute** as `cisco-av-pair`, **Type** as string, and the **Value** as shell:domain = <domain>/<role>/,<domain>// role as appropriate. For TACACS+, specify the **Attribute** as `cisco-av-pair`, **Requirement** as Mandatory, and the **Value** as shell:domain = <domain>/<role>/,<domain>// role as appropriate.

For example, if the *cisco-av-pair* is shell:domain = solar/admin/,common// read-all(16001), solar is the ACI tenant, admin is the role for this user that gives write privileges to this user in all of the tenant called solar, common is the ACI tenant common, read-all(16001) is the role with read privileges that gives this user read privileges to all of the ACI tenant common.

Step 5 Create a Service Selection Rule.

a) For RADIUS, create a service selection rule to associate the Identity Group with the Policy Element by navigating to **Access Policies** > **Default Device Network Access Identity** > **Authorization** and specifying the rule **Name**, **Status**, and **Conditions** as appropriate, and **Add** the `Internal Users:UserIdentityGroup in ALL Groups:<identity group name>`.

b) For TACACS+, create a service selection rule to associate the Identity Group with the Shell Profile by navigating to **Access Policies** > **Default Device Admin Identity** > **Authorization**. Specify the rule **Name**, **Conditions**, and **Select** the **Shell Profile** as appropriate.

What to Do Next

Use the newly created RADIUS and TACACS+ users to login to the APIC. Verify that the users have access to the correct APIC security domain according to the assigned RBAC roles and privileges. The users should not have access to items that have not been explicitly permitted. Read and write access rights should match those configured for that user.

Configuring Windows Server 2008 LDAP for APIC Access

Before You Begin

- First, configure the LDAP server, then configure the APIC for LDAP access.
- The Microsoft Windows Server 2008 is installed and online.
- The Microsoft Windows Server 2008 Server Manager ADSI Edit tool is installed. To install ADSI Edit, follow the instructions in the Windows Server 2008 Server Manager help.
- `AciCiscoAVPair` attribute specifications: Common Name = AciCiscoAVPair, LDAP Display Name = AciCiscoAVPair, Unique X500 Object ID = 1.3.6.1.4.1.9.22.1, Description = AciCiscoAVPair, Syntax = Case Sensitive String.

Note For LDAP configurations, best practice is to use AciCiscoAVPair as the attribute string. This avoids problems related to the limitation of common LDAP servers that do not allow overlapping object identifiers (OID); that is, the ciscoAVPair OID is already in use.

- A Microsoft Windows Server 2008 user account is available that will enable the following:
  - Running ADSI Edit to add the `AciCiscoAVPair` attribute to the Active Directory (AD) Schema.
  - Configuring an Active Directory LDAP user to have `AciCiscoAVPair` attribute permissions.

Procedure

Step 1 Log in to an Active Directory (AD) server as a domain administrator.

Step 2 Add the `AciCiscoAVPair` attribute to the AD schema.

a) Navigate to **Start** > **Run**, type mmc and press **Enter**.
The Microsoft Management Console (MMC) opens.
b) Navigate to **File** > **Add/Remove Sanp-in** > **Add**.
c) In the **Add Standalonee Snap-in** dialog box, select the **Active Directory Schema** and click **Add**.
The MMC **Console** opens.
d) Right-click the **Attributes** folder, select the **Create Attribute** option.
The **Create New Attribute** dialog box opens.
e) Enter AciCiscoAVPair for the **Common Name** , AciCiscoAVPair for the **LDAP Display Name**, 1.3.6.1.4.1.9.22.1 for the **Unique X500 Object ID**, and select Case Sensitive String for the **Syntax**.
f) Click **OK** to save the attribute.

Step 3 Update the **User Properties** class to include the **CiscoAVPair** attribute.

a) In the MMC **Console**, expand the **Classes** folder, right-click the **user** class, and choose **Properties**.
The **user Properties** dialog box opens.
b) Click the **Attributes** tab, select CiscoAVPair from the **Optional** list, and click **Add**.
The **Select Schema Object** dialog box opens.
c) In the **Select a schema object:** list, choose CiscoAVPair, and click **Apply**.
d) In the MMC **Console**, right-click the **Active Directory Schema**, and select **Reload the Schema**.

Step 4 Configure the AciCiscoAVPair attribute permissions.
Now that the LDAP includes the AciCiscoAVPair attributes, LDAP users need to be granted APIC permission by assigning them APIC RBAC roles.

a) In the ADSI Edit dialog box, locate a user who needs access to the APIC.
b) Right-click on the user name, and choose **Properties**.
The **<user> Properties** dialog box opens.
c) Click the **Attribute Editor** tab, select the *AciCiscoAVPair* attribute, and enter the *Value* as shell:domains = <domain>/<role>/,<domain>// role.
For example, if the *AciCiscoAVPair* is shell:domains = solar/admin/,common// read-all(16001), solar is the ACI tenant, admin is the role for this user that gives write privileges to this user in all of the tenant called solar, common is the ACI tenant common, read-all(16001) is the role with read privileges that gives this user read privileges to all of the ACI tennant common.
d) Click **OK** to save the changes and close the **<user> Properties** dialog box.

The LDAP server is configured to access the APIC.

What to Do Next

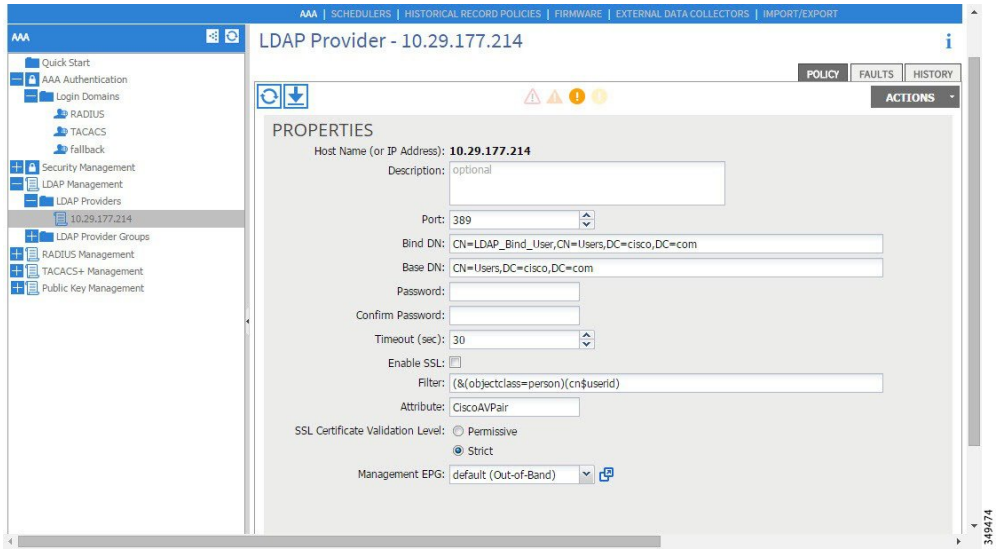
Configure the APIC for LDAP access.

Configuring APIC for LDAP Access

Before You Begin

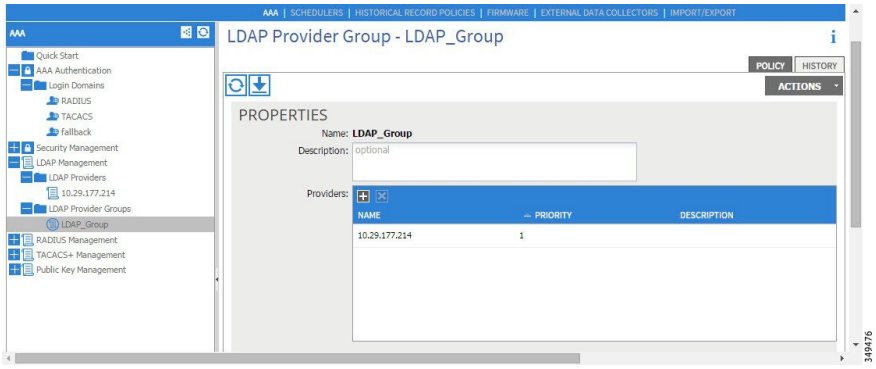
- The ACI fabric is installed, APIC controllers are online, and the APIC cluster is formed and healthy.
- The LDAP server host name or IP address, port, Bind DN, Base DN, and password are available.
- The APIC management EPG is available.
- An APIC user account is available that will enable the following:
  - Creating the LDAP provider and LDAP provider group.

Figure 7: LDAP Provider



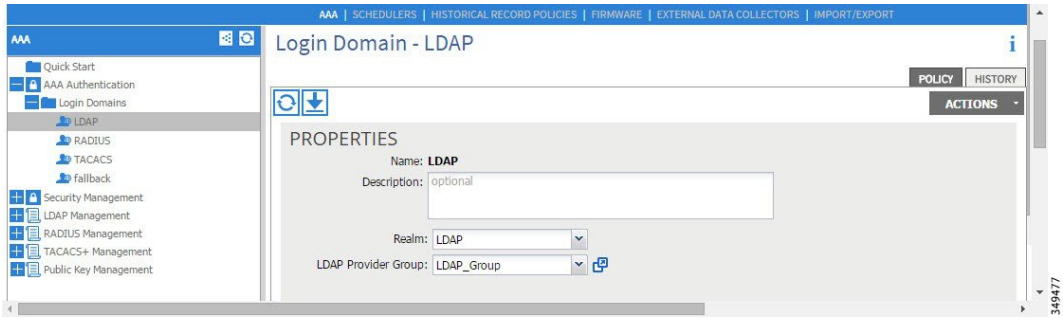
Note The Bind DN is the string that the APIC uses to log in to the LDAP server. The APIC uses this account to validate the remote user attempting to log in. The Base DN is the container name and path in the LDAP server where the APIC searches for the remote user account. This is where the password is validated. Filter is used to locate the attribute that the APIC requests to use for the *cisco-av-pair*. This contains the user authorization and assigned RBAC roles for use on the APIC. The APIC requests the attribute from the LDAP server.

Figure 8: LDAP Provider Group



◦ Creating the LDAP login domain.

Figure 9: AAA Login Domain for LDAP



Procedure

Step 1 On the APIC, configure the LDAP Provider.

a) On the APIC menu bar, click **ADMIN** > **AAA**.
b) In the **Navigation** pane, click the + icon to expand the**LDAP Managment** option.
c) In the **Navigation** pane, right-click the the**LDAP Providers** option, and select **Create LDAP Provider**.
d) Specify the LDAP host name (or IP address), port, Bind DN, Base DN, password, and management EPG.

Note If the APIC is configured for in-band management connectivity, choosing an out-of-band management EPG for LDAP access does not take effect. Alternatively, an out-of-band over an in-band management EPG can connect a LDAP server but requires configuring a static route for the LDAP server. The sample configuration procedures in this document use an APIC in-band management EPG.

Step 2 On the APIC, configure the LDAP Provider Group.

a) In the **Navigation** pane, right-click the the**LDAP Provider Groups** option, and select **Create LDAP Provider Group**

b) Specify the LDAP Provider Group name, description, and provider(s) as appropriate.

Step 3 On the APIC, configure the Login Domain for LDAP.

a) In the **Navigation** pane, click the + icon to expand the**AAA Authentication** option.

b) In the **Navigation** pane, right-click the **Login Domains** option, and select **Create Login Domain**.

c) Specify the Login Domain name, description, realm, and provider group as appropriate.

What to Do Next

This completes the APIC LDAP configuration steps. Next, test the APIC LDAP login access.

Changing the Default Behavior for Remote Users with Missing or Bad Cisco AV Pairs

Procedure

Step 1 On the menu bar, click **ADMIN** > **AAA**.

Step 2 In the **Navigation** pane, click **AAA Authentication**.

Step 3 In the **Work** pane, in the **Properties** area, from the **Remote user login policy** drop-down list, choose **Assign Default Role**.
The default value is **No Login**. The **Assign Default Role** option assigns the minimal read-only privileges to users that have missing or bad Cisco AV Pairs. Bad AV Pairs are those AV Pairs that fail the parsing rules.

Changing Default Behavior for Remote Users with Missing or Bad Cisco AV Pairs Using the NX-OS Style CLI

To change the default behavior for remote users with missing or bad Cisco AV pairs using the NX-OS CLI:

Procedure

Step 1 In the NX-OS CLI, start in Configuration mode.

Example:

```
apic1#
apic1# configure
```

Step 2 Configure the aaa user default role.

Example:

```
apic1(config)# aaa user default-role
 assign-default-role  assign-default-role
 no-login             no-login
```

Step 3 Configure the aaa authentication login methods.

Example:

```
apic1(config)# aaa authentication
 login  Configure methods for login

apic1(config)# aaa authentication login
 console  Configure console methods
 default  Configure default methods
 domain   Configure domain methods

apic1(config)# aaa authentication login console
 <CR>

apic1(config)# aaa authentication login domain
 WORD      Login domain name
 fallback
```

About Signature-Based Transactions

The APIC controllers in a Cisco ACI fabric offer different methods to authenticate users.

The primary authentication method uses a username and password and the APIC REST API returns an authentication token that can be used for future access to the APIC. This may be considered insecure in a situation where HTTPS is not available or enabled.

Another form of authentication that is offered utilizes a signature that is calculated for every transaction. The calculation of that signature uses a private key that must be kept secret in a secure location. When the APIC receives a request with a signature rather than a token, the APIC utilizes an X.509 certificate to verify the signature. In signature-based authentication, every transaction to the APIC must have a newly calculated signature. This is not a task that a user should do manually for each transaction. Ideally this function should be utilized by a script or an application that communicates with the APIC. This method is the most secure as it requires an attacker to crack the RSA/DSA key to forge or impersonate the user credentials.

Note Additionally, you must use HTTPS to prevent replay attacks.

Before you can use X.509 certificate-based signatures for authentication, verify that the following pre-requisite tasks are completed:

1 Create an X.509 certificate and private key using OpenSSL or a similar tool.

2 Create a local user on the APIC. (If a local user is already available, this task is optional).

3 Add the X.509 certificate to the local user on the APIC.

Guidelines and Limitations

Follow these guidelines and limitations:

- Local users are supported. Remote AAA users are not supported.
- The APIC GUI does not support the certificate authentication method.
- WebSockets and eventchannels do not work for X.509 requests.
- Certificates signed by a third party are not supported. Use a self-signed certificate.

Generating an X.509 Certificate and a Private Key

Procedure

Step 1 Enter an OpenSSL command to generate an X.509 certificate and private key.

Example:

```
$ openssl req -new -newkey rsa:1024 -days 36500 -nodes -x509 -keyout userabc.key -out
userabc.crt -subj '/CN=User ABC/O=Cisco Systems/C=US'
```

Note
- Once the X.509 certificate is generated, it will be added to the users profile on the APIC, and it is used to verify signatures. The private key is used by the client to generate the signatures.
- The certificate contains a public key but not the private key. The public key is the primary information used by the APIC to verify the calculated signature. The private key is never stored on the APIC. You must keep it secret.

Step 2 Display the fields in the certificate using OpenSSL.

Example:

```
$ openssl x509 -text -in userabc.crt
        Certificate:
            Data:
                Version: 3 (0x2)
                Serial Number:
                    c4:27:6c:4d:69:7c:d2:b6
                Signature Algorithm: sha1WithRSAEncryption
                Issuer: CN=User ABC, O=Cisco Systems, C=US
                Validity
                    Not Before: Jan 12 16:36:14 2015 GMT
                    Not After : Dec 19 16:36:14 2114 GMT
                Subject: CN=User ABC, O=Cisco Systems, C=US
                Subject Public Key Info:
                    Public Key Algorithm: rsaEncryption
                    RSA Public Key: (1024 bit)
                        Modulus (1024 bit):
                            00:92:35:12:cd:2b:78:ef:9d:ca:0e:11:77:77:3a:
                            99:d3:25:42:94:b5:3e:8a:32:55:ce:e9:21:2a:ff:
                            e0:e4:22:58:6d:40:98:b1:0d:42:21:db:cd:44:26:
                            50:77:e5:fa:b6:10:57:d1:ec:95:e9:86:d7:3c:99:
                            ce:c4:7f:61:1d:3c:9e:ae:d8:88:be:80:a0:4a:90:
```

```
                        d2:22:e9:1b:25:27:cd:7d:f3:a5:8f:cf:16:a8:e1:
                        3a:3f:68:0b:9c:7c:cb:70:b9:c7:3f:e8:db:85:d8:
                        98:f6:e3:70:4e:47:e2:59:03:49:01:83:8e:50:4a:
                        5f:bc:35:d2:b1:07:be:ec:e1
                    Exponent: 65537 (0x10001)
            X509v3 extensions:
                X509v3 Subject Key Identifier:
                    0B:E4:11:C7:23:46:10:4F:D1:10:4C:C1:58:C2:1E:18:E8:6D:85:34
                X509v3 Authority Key Identifier:
                    keyid:0B:E4:11:C7:23:46:10:4F:D1:10:4C:C1:58:C2:1E:18:E8:6D:85:34
                    DirName:/CN=User ABC/O=Cisco Systems/C=US
                    serial:C4:27:6C:4D:69:7C:D2:B6

                X509v3 Basic Constraints:
                    CA:TRUE
        Signature Algorithm: sha1WithRSAEncryption
            8f:c4:9f:84:06:30:59:0c:d2:8a:09:96:a2:69:3d:cf:ef:79:
            91:ea:cd:ae:80:16:df:16:31:3b:69:89:f7:5a:24:1f:fd:9f:
            d1:d9:b2:02:41:01:b9:e9:8d:da:a8:4c:1e:e5:9b:3e:1d:65:
            84:ff:e8:ad:55:3e:90:a0:a2:fb:3e:3e:ef:c2:11:3d:1b:e6:
            f4:5e:d2:92:e8:24:61:43:59:ec:ea:d2:bb:c9:9a:7a:04:91:
            8e:91:bb:9d:33:d4:28:b5:13:ce:dc:fe:c3:e5:33:97:5d:37:
            cc:5f:ad:af:5a:aa:f4:a3:a8:50:66:7d:f4:fb:78:72:9d:56:
            91:2c
    [snip]
```

Configuring a Local User

Creating a Local User and Adding a User Certificate Using the GUI

Procedure

Step 1 On the menu bar, choose **ADMIN** > **AAA**.

Step 2 In the **Navigation** pane, click **AAA Authentication**.

Step 3 In the **Work** pane, verify that in the default **Authentication** field, the **Realm** field displays as Local.

Step 4 In the **Navigation** pane, expand **Security Management** > **Local Users**.
The admin user is present by default.

Step 5 In the **Navigation** pane, right-click **Local Users** and click **Create Local User**.

Step 6 In the **Security** dialog box, choose the desired security domain for the user, and click **Next**.

Step 7 In the **Roles** dialog box, click the radio buttons to choose the roles for your user, and click **Next**.
You can provide read-only or read/write privileges.

Step 8 In the **User Identity** dialog box, perform the following actions:

a) In the **Login ID** field, add an ID.
b) In the **Password** field, enter the password.
c) In the **Confirm Password** field, confirm the password.
d) Click **Finish**.

Step 9 In the **Navigation** pane, click the name of the user that you created. In the **Work** pane, expand the + sign next to your user in the **Security Domains** area.

The access privileges for your user are displayed.

Step 10 In the **Work** pane, in the **User Certificates** area, click the user certificates + sign, and in the **Create X509 Certificate** dialog box, perform the following actions:

a) In the **Name** field, enter a certificate name.
b) In the **Data** field, enter the user certificate details.
c) Click **Submit**.

The X509 certificate is created for the local user.

Creating a Local User and Adding a User Certificate Using the REST API

Procedure

Create a local user and add a user certificate.

Example:

```
method: POST
        url: http://apic/api/node/mo/uni/userext/user-userabc.json
        payload:
        {
            "aaaUser": {
                "attributes": {
                    "name": "userabc",
                    "firstName": "Adam",
                    "lastName": "BC",
                    "phone": "408-525-4766",
                    "email": "userabc@cisco.com",
                },
                "children": [{
                    "aaaUserCert": {
                        "attributes": {
                            "name": "userabc.crt",
                          "data": "-----BEGIN CERTIFICATE-----\nMIICjjCCAfegAwIBAgIJAMQnbE
 <snipped content> ==\n-----END CERTIFICATE-----",
                        },
                        "children": []
                    },
                    "aaaUserDomain": {
                        "attributes": {
                            "name": "all",
                        },
                        "children": [{
                            "aaaUserRole": {
                                "attributes": {
                                    "name": "aaa",
                                    "privType": "writePriv",
                                },
                                "children": []
                            }
                        }, {
                            "aaaUserRole": {
                                "attributes": {
                                    "name": "access-admin",
                                    "privType": "writePriv",
                                },
                                "children": []
                            }
                        }, {
                            "aaaUserRole": {
                                "attributes": {
```

```
                                    "name": "admin",
                                    "privType": "writePriv",
                                },
                                "children": []
                            }
                        }, {
                            "aaaUserRole": {
                                "attributes": {
                                    "name": "fabric-admin",
                                    "privType": "writePriv",
                                },
                                "children": []
                            }
                        }, {
                            "aaaUserRole": {
                                "attributes": {
                                    "name": "nw-svc-admin",
                                    "privType": "writePriv",
                                },
                                "children": []
                            }
                        }, {
                            "aaaUserRole": {
                                "attributes": {
                                    "name": "ops",
                                    "privType": "writePriv",
                                },
                                "children": []
                            }
                        }, {
                            "aaaUserRole": {
                                "attributes": {
                                    "name": "read-all",
                                    "privType": "writePriv",
                                },
                                "children": []
                            }
                        }, {
                            "aaaUserRole": {
                                "attributes": {
                                    "name": "tenant-admin",
                                    "privType": "writePriv",
                                },
                                "children": []
                            }
                        }, {
                            "aaaUserRole": {
                                "attributes": {
                                    "name": "tenant-ext-admin",
                                    "privType": "writePriv",
                                },
                                "children": []
                            }
                        }, {
                            "aaaUserRole": {
                                "attributes": {
                                    "name": "vmm-admin",
                                    "privType": "writePriv",
                                },
                                "children": []
                            }
                        }]
                    }
                }]
            }
        }
```

Creating a Local User Using Python SDK

Procedure

Create a local user.

Example:

```
#!/usr/bin/env python
from cobra.model.pol import Uni as PolUni
from cobra.model.aaa import UserEp as AaaUserEp
from cobra.model.aaa import User as AaaUser
from cobra.model.aaa import UserCert as AaaUserCert
from cobra.model.aaa import UserDomain as AaaUserDomain
from cobra.model.aaa import UserRole as AaaUserRole
from cobra.mit.access import MoDirectory
from cobra.mit.session import LoginSession
from cobra.internal.codec.jsoncodec import toJSONStr

APIC = 'http://10.10.10.1'
username = 'admin'
password = 'p@$$w0rd'

session = LoginSession(APIC, username, password)
modir = MoDirectory(session)
modir.login()

def readFile(fileName=None, mode="r"):
   if fileName is None:
      return ""
   fileData = ""
   with open(fileName, mode) as aFile:
      fileData = aFile.read()
   return fileData

# Use a dictionary to define the domain and a list of tuples to define
# our aaaUserRoles (roleName, privType)
# This can further be abstracted by doing a query to get the valid
# roles, that is what the GUI does

userRoles = {'all': [
               ('aaa', 'writePriv'),
               ('access-admin', 'writePriv'),
               ('admin', 'writePriv'),
               ('fabric-admin', 'writePriv'),
               ('nw-svc-admin', 'writePriv'),
               ('ops', 'writePriv'),
               ('read-all', 'writePriv'),
               ('tenant-admin', 'writePriv'),
               ('tenant-ext-admin', 'writePriv'),
               ('vmm-admin', 'writePriv'),
           ],
}

uni = PolUni('') # '' is the Dn string for topRoot
aaaUserEp = AaaUserEp(uni)
aaaUser = AaaUser(aaaUserEp, 'userabc', firstName='Adam',
                  email='userabc@cisco.com')

aaaUser.lastName = 'BC'
aaaUser.phone = '555-111-2222'
aaaUserCert = AaaUserCert(aaaUser, 'userabc.crt')
aaaUserCert.data = readFile("/tmp/userabc.crt")
# Now add each aaaUserRole to the aaaUserDomains which are added to the
# aaaUserCert
for domain,roles in userRoles.items():
   aaaUserDomain = AaaUserDomain(aaaUser, domain)
```

```
   for roleName, privType in roles:
       aaaUserRole = AaaUserRole(aaaUserDomain, roleName,
                                 privType=privType)
print toJSONStr(aaaUser, prettyPrint=True)

cr = ConfigRequest()
cr.addMo(aaaUser)
modir.commit(cr)
# End of Script to create a user
```

Using a Private Key to Calculate a Signature

Before You Begin

You must have the following information available:

- HTTP method - GET, POST, DELETE
- REST API URI being requested, including any query options
- For POST requests, the actual payload being sent to the APIC
- The private key used to generate the X.509 certificate for the user
- The distinguished name for the user X.509 certificate on the APIC

Procedure

Step 1 Concatenate the HTTP method, REST API URI, and payload together in this order and save them to a file. This concatenated data must be saved to a file for OpenSSL to calculate the signature. In this example, we use a filename of payload.txt. Remember that the private key is in a file called userabc.key.

Example:
GET example:
```
GET http://10.10.10.1/api/class/fvTenant.json?rsp-subtree=children
```
POST example:
```
POST http://10.10.10.1/api/mo/tn-test.json{"fvTenant": {"attributes": {"status": "deleted",
 "name": "test"}}}
```

Step 2 Calculate a signature using the private key and the payload file using OpenSSL.

Example:
```
openssl dgst -sha256 -sign userabc.key payload.txt > payload_sig.bin
```
The resulting file has the signature printed on multiple lines.

Step 3 Strip the signature of the new lines using Bash.

Example:
```
$ tr -d '\n' < payload_sig.base64
P+OTqK0CeAZjl7+Gute2R1Ww8OGgtzE0wsLlx8fIXXl4V79Zl7
Ou8IdJH9CB4W6CEvdICXqkv3KaQszCIC0+Bn07o3qF//BsIplZmYChD6gCX3f7q
IcjGX+R6HAqGeK7k97cNhXlWEoobFPe/oajtPjOu3tdOjhf/9ujG6Jv6Ro=
```
Note This is the signature that will be sent to the APIC for this specific request. Other requests will require to have their own signatures calculated.

Step 4 Place the signature inside a string to enable the APIC to verify the signature against the payload.

This complete signature is sent to the APIC as a cookie in the header of the request.

Example:

```
APIC-Request-Signature=P+OTqK0CeAZjl7+Gute2R1Ww8OGgtzE0wsLlx8f
IXXl4V79Zl7Ou8IdJH9CB4W6CEvdICXqkv3KaQszCIC0+Bn07o3qF//BsIplZmYChD6gCX3f
7qIcjGX+R6HAqGeK7k97cNhXlWEoobFPe/oajtPjOu3tdOjhf/9ujG6Jv6Ro=;
APIC-Certificate-Algorithm=v1.0; APIC-Certificate-Fingerprint=fingerprint;
APIC-Certificate-DN=uni/userext/user-userabc/usercert-userabc.crt
```

Note The DN used here must match the DN of the user certified object containing the x509 certificate in the next step.

Step 5 Use the CertSession class in the Python SDK to communicate with an APIC using signatures.
The following script is an example of how to use the CertSession class in the ACI Python SDK to make requests to an APIC using signatures.

Example:

```
#!/usr/bin/env python
# It is assumed the user has the X.509 certificate already added to
# their local user configuration on the APIC
from cobra.mit.session import CertSession
from cobra.mit.access import MoDirectory

def readFile(fileName=None, mode="r"):
    if fileName is None:
        return ""
    fileData = ""
    with open(fileName, mode) as aFile:
        fileData = aFile.read()
    return fileData

pkey = readFile("/tmp/userabc.key")
csession = CertSession("https://ApicIPOrHostname/",
                       "uni/userext/user-userabc/usercert-userabc.crt", pkey)

modir = MoDirectory(csession)
resp = modir.lookupByDn('uni/fabric')
pring resp.dn
# End of script
```

Note The DN used in the earlier step must match the DN of the user certified object containing the x509 certificate in this step.

Accounting

ACI fabric accounting is handled by these two managed objects (MO) that are processed by the same mechanism as faults and events:

- The `aaaSessionLR` MO tracks user account login/log-out sessions on the APIC and switches, and token refresh. The ACI fabric session alert feature stores information such as the following:
  - Username
  - IP address initiating the session
  - Type (telnet, https, REST etc.)
  - Session time and length

- Token refresh – a user account login event generates a valid active token which is required in order for the user account to exercise its rights in the ACI fabric.

Note Token expiration is independent of login; a user could log out but the token expires according to the duration of the timer value it contains.

- The `aaaModLR` MO tracks the changes users make to objects and when the changes occurred.

Both `aaaSessionLR` and `aaaModLR` event logs are stored in APIC shards. Once the data exceeds the pre-set storage allocation size, it overwrites records on a first-in first-out basis.

Note In the event of a destructive event such as a disk crash or a fire that destroys an APIC cluster node, the event logs are lost; event logs are not replicated across the cluster.

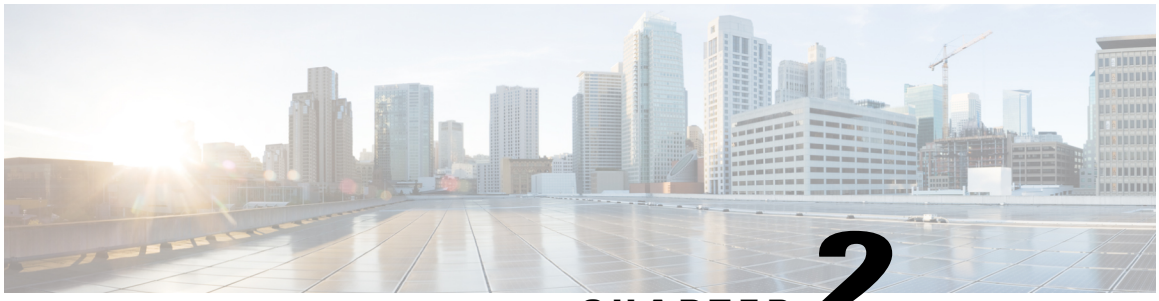
The `aaaModLR` and `aaaSessionLR` MOs can be queried by class or by distinguished name (DN). A class query will give you all the log records for the whole fabric. All `aaaModLR` records for the whole fabric are available from the GUI at the Fabric -> Inventory -> pod-1 -> history -> audit log section, The GUI => History => Log options enable viewing event logs for a specific object identified in the GUI context.

The standard syslog, callhome, REST query, and CLI export mechanism are fully supported for aaaModLR and aaaSessionLR MO query data. There is no default policy to export this data.

There are no pre-configured queries in the APIC that report on aggregations of data across a set of objects or for the entire system. A fabric administrator can configure export policies that periodically export `aaaModLR` and `aaaSessionLR` query data to a syslog server. Exported data can be archived periodically and used to generate custom reports from portions of the system or across the entire set of system logs.

Routed Connectivity to External Networks as a Shared Service Billing and Statistics

The APIC can be configured to collect byte count and packet count billing statistics from a port configured for routed connectivity to external networks (an `l3extInstP` EPG) as a shared service. Any EPG in any tenant can share an `l3extInstP` EPG for routed connectivity to external networks. Billing statistics can be collected for each EPG in any tenant that uses an `l3extInstP` EPG as a shared service. The leaf switch where the `l3extInstP` is provisioned forwards the billing statistics to the APIC where they are aggregated. Accounting policies can be configured to periodically export these billing statics to a server.



CHAPTER 2

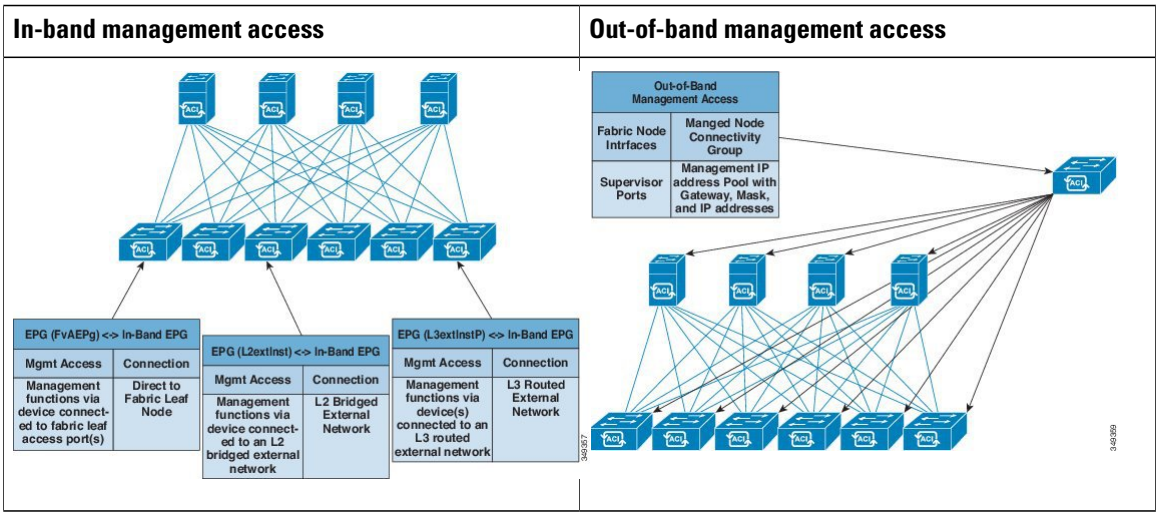
Management

This chapter contains the following sections:

Management Workflows

ACI Management Access Workflows

This workflow provides an overview of the steps required to configure management connectivity to switches in the ACI fabric.



1. Prerequisites

- Ensure that you have read/write access privileges to the infra security domain.
- Ensure that the target leaf switches with the necessary interfaces are available.

2. Configure the ACI Leaf Switch Access Ports

Choose which of these management access scenarios you will use:

- For **in-band** management, follow the suggested topics for in-band configuration.
- For **out-of-band** management, follow the suggested topics for out-of-band configuration.

Suggested topics

For additional information, see the following topics:

- Configuring In-Band Management Access Using the Advanced GUI, on page 29
- Configuring In-Band Management Access Using the NX-OS Style CLI, on page 33
- Configuring In-Band Management Access Using the REST API, on page 33
- Configuring Out-of-Band Management Access Using the Advanced GUI, on page 36
- Configuring Out-of-Band Management Access Using the NX-OS Style CLI, on page 38
- Configuring Out-of-Band Management Access Using the REST API, on page 38

Adding Management Access

An APIC controller has two routes to reach the management network, one is by using the in-band management interface and the other is by using the out-of-band management interface.

- In-band management access—You can configure in-band management connectivity to the APIC and the ACI fabric. You first configure the VLANs that will be used by APIC when the APIC is communicating with the leaf switches, and then you configure the VLANs that the VMM servers will use to communicate with the leaf switches.
- Out-of-band management access—You can configure out-of-band management connectivity to the APIC and the ACI fabric. You configure an out-of-band contract that is associated with an out-of-band endpoint group (EPG), and attach the contract to the external network profile.

Note The APIC out-of-band management connection link must be 1 Gbps.

The APIC controller always selects the in-band management interface over the out-of-band management interface, if the in-band management interface is configured. The out-of-band management interface is used only when the in-band management interface is not configured, or if the destination address is on the same subnet as the out-of-band management subnet of the APIC. This behavior cannot be changed or reconfigured.

The APIC management interface does not support an IPv6 address and cannot connect to an external IPv6 server through this interface.

Configuring the external management instance profile under the management tenant for in-band or out-of-band has no effect on the protocols that are configured under the fabric-wide communication policies. The subnets and contracts specified under the external management instance profile do not affect HTTP/HTTPS or SSH/Telnet.

In-Band and Out-of-Band Management Access

The mgmt tenant provides a convenient means to configure access to fabric management functions. While fabric management functions are accessible through the APIC, they can also be accessed directly through in-band and out-of-band network policies.

Configuring In-Band Management Access Using the Advanced GUI

Note

- IPv4 and IPv6 addresses are supported for in-band management access. IPv6 configurations are supported using static configurations (for both in-band and out-of-band). IPv4 and IPv6 dual in-band and out-of-band configurations are supported only through static configuration. For more information, see the KB article,*Configuring Static Management Access in Cisco APIC*.
- To watch an example video of this task, see Videos Webpage.

Procedure

Step 1 On the menu bar, choose **FABRIC** > **Access Policies**. In the **Navigation** pane, expand **Interface Policies**.

Step 2 In the **Navigation** pane, right-click **Switch Policies** and choose **Configure Interface, PC and VPC**.

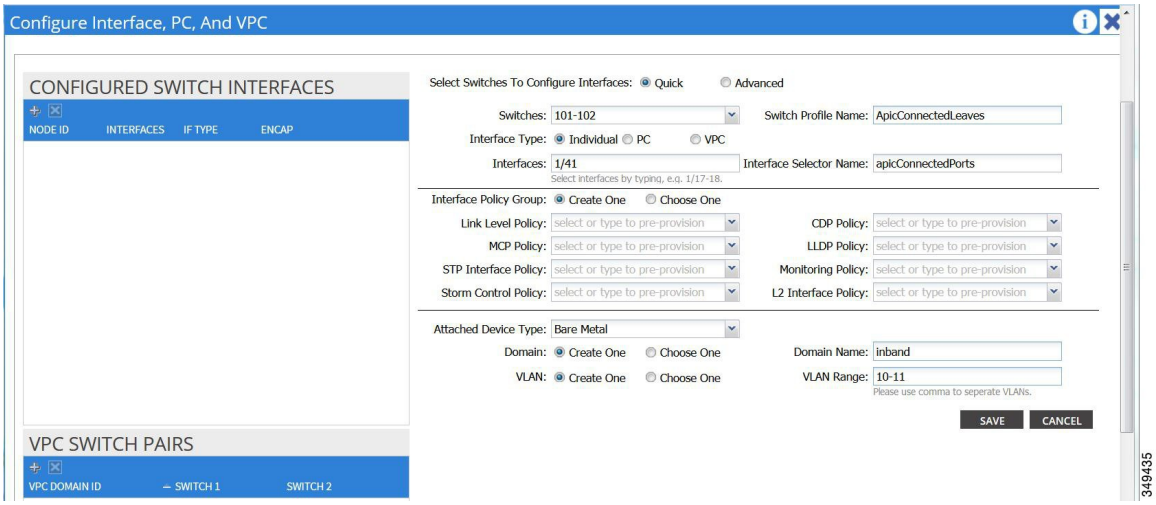
Step 3 In the **Configure Interface, PC, and VPC** dialog box, to configure switch ports connected to APICs, perform the following actions:

a) Click the large + icon next to the switch diagram to create a new profile and configure VLANs for the APIC.

b) From the **Switches** field drop-down list, check the check boxes for the switches to which the APICs are connected. (leaf1 and leaf2).

c) In the **Switch Profile Name** field, enter a name for the profile (apicConnectedLeaves).

d) Click the + icon to configure the ports.
A dialog box similar to the following image is displayed for the user to enter the content:



e) Verify that in the **Interface Type** area, the **Individual** radio button is selected.

f) In the **Interfaces** field, enter the ports to which APICs are connected.

g) In the **Interface Selector Name** field, enter the name of the port profile (apicConnectedPorts).

h) In the **Interface Policy Group** field, click the **Create One** radio button.

i) In the **Attached Device Type** field, choose the appropriate device type to configure the domain (Bare Metal).

j) In the **Domain** field, click the **Create One** radio button.

k) In the **Domain Name** field, enter the domain name. (**inband**)

l) In the **VLAN** field, choose the **Create One** radio button.

m) In the **VLAN Range** field, enter the VLAN range. Click **Save**, and click **Save** again. Click **Submit**.

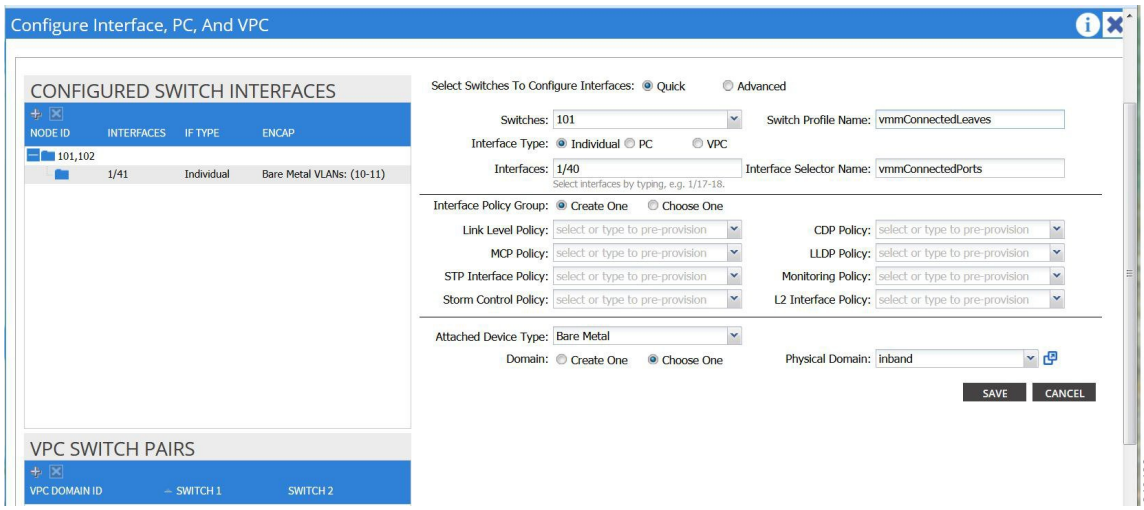
Step 4 In the **Navigation** pane, right-click **Switch Policies** and choose **Configure Interface, PC and VPC**.

Step 5 In the **Configure Interface, PC, and VPC** dialog box, perform the following actions:

a) Click the large + icon next to the switch diagram to create a new profile and configure VLANs for the server.

b) In the **Switches** field, from drop-down list, check the check boxes for the switches to which the servers are connected. (leaf1).

c) In the **Switch Profile Name** field, enter a name for the profile (vmmConnectedLeaves).
d) Click the + icon to configure the ports.
A dialog box similar to the following image is displayed for the user to enter the content:



e) Verify that in the **Interface Type** area, the **Individual** radio button is selected.
f) In the **Interfaces** field, enter the ports to which the servers are connected (1/40).
g) In the **Interface Selector Name** field, enter the name of the port profile.
h) In the **Interface Policy Group** field, click the **Create One** radio button.
i) In the **Attached Device Type** field, choose the appropriate device type to configure the domain (Bare Metal).
j) In the **Domain** field, from the drop-down list click the **Choose One** radio button
k) From the **Physical Domain** drop-down list, choose the domain created earlier.
l) In the **Domain Name** field, enter the domain name.
m) Click **Save**, and click **Save** again.

Step 6 In the **Configure Interface, PC, and VPC** dialog box, click **Submit**.

Step 7 On the menu bar, click **TENANTS** > **mgmt**. In the **Navigation** pane, expand **Tenant mgmt** > **Networking** > **Bridge Domains** to configure the bridge domain on the in-band connection.

Step 8 Expand the in-band bridge domain (inb). Right-click **Subnets**. Click **Create Subnet** and perform the following actions to configure the in-band gateway:

a) In the **Create Subnet** dialog box, in the **Gateway IP** field, enter the in-band management gateway IP address and mask.
b) Click **Submit**.

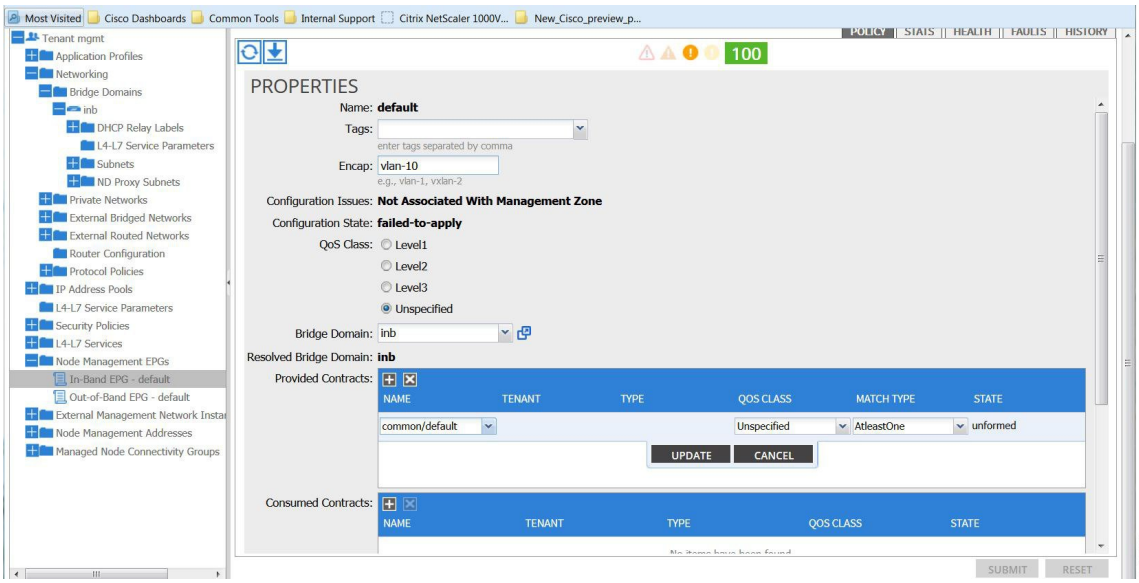
Step 9 In the **Navigation** pane, expand **Tenant mgmt** > **Node Management EPGs**. Right-click **Node Management EPGs** and choose **Create In-Band Management EPG**. Perform the following actions to set the VLAN on the in-band EPG used to communicate with the APIC:

a) In the Name field, enter the in-band management EPG name.
b) In the **Encap** field, enter the VLAN (vlan-10).
c) From the **Bridge Domain** drop-down field, choose the bridge domain. Click **Submit**.
d) In the **Navigation** pane, choose the newly created in-band EPG.

e) Expand **Provided Contracts**. In the **Name** field, from the drop-down list, choose the default contract to enable EPG to provide the default contract that will be consumed by the EPGs on which the VMM servers are located.

f) Click **Update**, and click **Submit**.

A dialog box similar to the following image is displayed:



Step 10 In the **Navigation** pane, right-click **Node Management Addresses** and click **Create Node Management Addresses**, and perform the following actions to configure the IP addresses to be assigned to APIC controllers in the fabric:

a) In the **Create Node Management Addresses** dialog box, in the **Policy Name** field, enter the policy name (apicInb).

b) In the **Nodes** field, **Select** column, check the check boxes for the nodes that will be part of this fabric (apic1, apic2, apic3).

c) In the **Config** field, check the **In-Band Addresses** check box.

d) In the **Node Range** fields, enter the range.

e) In the **In-Band IP Addresses** area, in the **In-Band Management EPG** field, from the drop-down list, choose default. This associates the default in-band Management EPG.

f) In the **In-Band IP Addresses** and **Gateway** fields, enter the IPv4 or IPv6 addresses as desired.

g) Click **Submit**. The IP addresses for the APICs are now configured.

Step 11 In the **Navigation** pane, right-click **Node Management Addresses**. Click **Create Node Management Addresses**, and perform the following actions to configure the IP addresses for the leaf and spine switches in the fabric:

a) In the **Create Node Management Addresses** dialog box, in the **Policy Name** field, enter the policy name (switchInb).

b) In the **Nodes** field, **Select** column, check the check boxes next to the nodes that will be part of this fabric (leaf1, leaf2, spine1, spine2).

c) In the **Config** field, click the **In-Band Addresses** checkbox.

d) In the **Node Range** fields, enter the range.

e) In the **In-Band IP Addresses** area, in the **In-Band Management EPG** field, from the drop-down list, choose default. The default in-band management EPG is now associated.

f) In the **In-Band IP Addresses** and **Gateway** fields, enter the IPv4 or IPv6 addresses as desired.
g) Click **Submit**. In the **Confirm** dialog box, click **Yes**. The IP addresses for the leaf and spine switches are now configured.

Step 12 In the **Navigation** pane, under **Node Management Addresses**, click the APIC policy name (apicInb) to verify the configurations. In the **Work** pane, the IP addresses assigned to various nodes are displayed.

Step 13 In the **Navigation** pane, under **Node Management Addresses**, click the switches policy name (switchInb). In the **Work** pane, the IP addresses that are assigned to switches and the gateway addresses they are using are displayed.

Configuring In-Band Management Access Using the NX-OS Style CLI

Procedure

Step 1 Assign a VLAN for the APIC inband management, as shown in the following example:

Example:

```
apic1(config)#
apic1(config)# vlan-domain inband-mgmt
apic1(config-vlan) vlan 10
apic1(config-vlan) exit
```

Step 2 Provide external connectivity to the inband management ports, as shown in the following example:

Example:

Note In this step, the controller is connected to a port on a leaf switch. You must add a VLAN domain member on that port. In this example, in leaf 101, the port ethernet 1/2 is connected to controller 1. You are configuring the VLAN domain member "inband management". This is one part of the connection. The other part is that the management station is connected to leaf 102, interface ethernet 1/3. A controller is one machine connected one port on the leaf switch, which in this case is leaf 102. The machine is trying to connect to the controller from the outside (ethernet 1/3).

```
apic1(config)#
apic1(config)# leaf 101
apic1(config-leaf) internet ethernet 1/2
apic1(config-leaf-if)# vlan-domain member inband-mgmt
apic1(config-leaf-if)# exit
apic1(config)# leaf 102
apic1(config-leaf) internet ethernet 1/3
apic1(config-leaf-if)# vlan-domain member inband-mgmt
apic1(config-leaf-if)# switchport trunk allowed vlan
apic1(config-leaf-if)# exit
```

Configuring In-Band Management Access Using the REST API

IPv4 and IPv6 addresses are supported for in-band management access. IPv6 configurations are supported using static configurations (for both in-band and out-of-band). IPv4 and IPv6 dual in-band and out-of-band

configurations are supported only through static configuration. For more information, see the KB article,*Configuring Static Management Access in Cisco APIC*.

Procedure

Step 1 Create a VLAN namespace.

Example:

```
POST
https://APIC-IP/api/mo/uni.xml

<?xml version="1.0" encoding="UTF-8"?>
<!-- api/policymgr/mo/uni.xml -->
<polUni>
  <infraInfra>
    <!-- Static VLAN range -->
    <fvnsVlanInstP name="inband" allocMode="static">
      <fvnsEncapBlk name="encap" from="vlan-10" to="vlan-11"/>
    </fvnsVlanInstP>
  </infraInfra>
</polUni>
```

Step 2 Create a physical domain.

Example:

```
POST
https://APIC-IP/api/mo/uni.xml

<?xml version="1.0" encoding="UTF-8"?>
<!-- api/policymgr/mo/uni.xml -->
<polUni>
  <physDomP name="inband">
    <infraRsVlanNs tDn="uni/infra/vlanns-inband-static"/>
  </physDomP>
</polUni>
```

Step 3 Create selectors for the in-band management.

Example:

```
POST
https://APIC-IP/api/mo/uni.xml

<?xml version="1.0" encoding="UTF-8"?>
<!-- api/policymgr/mo/.xml -->
<polUni>
  <infraInfra>
    <infraNodeP name="vmmNodes">
      <infraLeafS name="leafS" type="range">
        <infraNodeBlk name="single0" from_="101" to_="101"/>
      </infraLeafS>
      <infraRsAccPortP tDn="uni/infra/accportprof-vmmPorts"/>
    </infraNodeP>

    <!-- Assumption is that VMM host is reachable via eth1/40. -->
    <infraAccPortP name="vmmPorts">
      <infraHPortS name="portS" type="range">
        <infraPortBlk name="block1"
                      fromCard="1" toCard="1"
                      fromPort="40" toPort="40"/>
        <infraRsAccBaseGrp tDn="uni/infra/funcprof/accportgrp-inband" />
      </infraHPortS>
    </infraAccPortP>


    <infraNodeP name="apicConnectedNodes">
      <infraLeafS name="leafS" type="range">
```

```
        <infraNodeBlk name="single0" from_="101" to_="102"/>
      </infraLeafS>
      <infraRsAccPortP tDn="uni/infra/accportprof-apicConnectedPorts"/>
    </infraNodeP>

    <!-- Assumption is that APIC is connected to eth1/1. -->
    <infraAccPortP name="apicConnectedPorts">
      <infraHPortS name="portS" type="range">
        <infraPortBlk name="block1"
                      fromCard="1" toCard="1"
                      fromPort="1" toPort="3"/>
        <infraRsAccBaseGrp tDn="uni/infra/funcprof/accportgrp-inband" />
      </infraHPortS>
    </infraAccPortP>

    <infraFuncP>
      <infraAccPortGrp name="inband">
        <infraRsAttEntP tDn="uni/infra/attentp-inband"/>
      </infraAccPortGrp>
    </infraFuncP>

    <infraAttEntityP name="inband">
      <infraRsDomP tDn="uni/phys-inband"/>
    </infraAttEntityP>
  </infraInfra>
</polUni>
```

Step 4 Configure an in-band bridge domain and endpoint group (EPG).

Example:

```
POST
https://APIC-IP/api/mo/uni.xml

<?xml version="1.0" encoding="UTF-8"?>
<!-- api/policymgr/mo/.xml -->
<polUni>
  <fvTenant name="mgmt">
    <!-- Configure the in-band management gateway address on the
         in-band BD. -->
    <fvBD name="inb">
      <fvSubnet ip="10.13.1.254/24"/>
    </fvBD>

    <mgmtMgmtP name="default">
      <!-- Configure the encap on which APICs will communicate on the
           in-band network. -->
      <mgmtInB name="default" encap="vlan-10">
        <fvRsProv tnVzBrCPName="default"/>
      </mgmtInB>
    </mgmtMgmtP>
  </fvTenant>
</polUni>
```

Step 5 Create an address pool.

Example:

```
POST
https://APIC-IP/api/mo/uni.xml

<?xml version="1.0" encoding="UTF-8"?>
<!-- api/policymgr/mo/.xml -->
<polUni>
  <fvTenant name="mgmt">
    <!-- Adresses for APIC in-band management network -->
    <fvnsAddrInst name="apicInb" addr="10.13.1.254/24">
      <fvnsUcastAddrBlk from="10.13.1.1" to="10.13.1.10"/>
    </fvnsAddrInst>

    <!-- Adresses for switch in-band management network -->
    <fvnsAddrInst name="switchInb" addr="10.13.1.254/24">
```

```
      <fvnsUcastAddrBlk from="10.13.1.101" to="10.13.1.120"/>
    </fvnsAddrInst>
  </fvTenant>
</polUni>
```

Note Dynamic address pools for IPv6 is not supported.

Step 6 Create management groups.

Example:

```
POST
https://APIC-IP/api/mo/uni.xml

<?xml version="1.0" encoding="UTF-8"?>
<!-- api/policymgr/mo/.xml -->
<polUni>
  <infraInfra>
    <!-- Management node group for APICs -->
    <mgmtNodeGrp name="apic">
      <infraNodeBlk name="all" from_="1" to_="3"/>
      <mgmtRsGrp tDn="uni/infra/funcprof/grp-apic"/>
    </mgmtNodeGrp>

    <!-- Management node group for switches-->
    <mgmtNodeGrp name="switch">
      <infraNodeBlk name="all" from_="101" to_="104"/>
      <mgmtRsGrp tDn="uni/infra/funcprof/grp-switch"/>
    </mgmtNodeGrp>

    <!-- Functional profile -->
    <infraFuncP>
      <!-- Management group for APICs -->
      <mgmtGrp name="apic">
        <!-- In-band management zone -->
        <mgmtInBZone name="default">
          <mgmtRsInbEpg tDn="uni/tn-mgmt/mgmtp-default/inb-default"/>
          <mgmtRsAddrInst tDn="uni/tn-mgmt/addrinst-apicInb"/>
        </mgmtInBZone>
      </mgmtGrp>

      <!-- Management group for switches -->
      <mgmtGrp name="switch">
        <!-- In-band management zone -->
        <mgmtInBZone name="default">
          <mgmtRsInbEpg tDn="uni/tn-mgmt/mgmtp-default/inb-default"/>
          <mgmtRsAddrInst tDn="uni/tn-mgmt/addrinst-switchInb"/>
        </mgmtInBZone>
      </mgmtGrp>
    </infraFuncP>
  </infraInfra>
</polUni>
```

Note Dynamic address pools for IPv6 is not supported.

Configuring Out-of-Band Management Access Using the Advanced GUI

Note

- IPv4 and IPv6 addresses are supported for out-of-band management access.
- To watch an example video of this task, see Videos Webpage.

Before You Begin

The APIC out-of-band management connection link must be 1 Gbps.

Procedure

Step 1 On the menu bar, choose **TENANTS** > **mgmt**. In the **Navigation** pane, expand **Tenant mgmt**.

Step 2 Right-click **Node Management Addresses**, and click **Create Node Management Addresses**.

Step 3 In the **Create Node Management Addresses** dialog box, perform the following actions:

a) In the **Policy Name** field, enter a policy name (switchOob).

b) In the **Nodes** field, check the check boxes next to the appropriate leaf and spine switches (leaf1, leaf2, spine1).

c) In the **Config** field, check the check box for **Out of-Band Addresses**.

Note The **Out-of-Band IP addresses** area is displayed.

d) In the **Out-of-Band Management EPG** field, choose the EPG from the drop-down list (default).

e) In the **Out-of-Band IP Addresses** and Out-of-Band Gateway fields, enter the desired IPv4 or IPv6 addresses that will be assigned to the switches. Click **OK**.

The node management IP addresses are configured. You must configure out-of-band management access addresses for the leaf and spine switches as well as for APIC.

Step 4 In the **Navigation** pane, expand **Node Management Addresses**, and click the policy that you created.
In the **Work** pane, the out-of-band management addresses are displayed against the switches.

Step 5 In the **Navigation** pane, expand **Security Policies** > **Out-of-Band Contracts**.

Step 6 Right-click **Out-of-Band Contracts**, and click **Create Out-of-Band Contract**.

Step 7 In the **Create Out-of-Band Contract** dialog box, perform the following tasks:

a) In the **Name** field, enter a name for the contract (oob-default).

b) Expand **Subjects**. In the **Create Contract Subject** dialog box, in the **Name** field, enter a subject name (oob-default).

c) Expand **Filters**, and in the **Name** field, from the drop-down list, choose the name of the filter (default). Click **Update**, and click **OK**.

d) In the **Create Out-of-Band Contract** dialog box, click **Submit**.

An out-of-band contract that can be applied to the out-of-band EPG is created.

Step 8 In the **Navigation** pane, expand **Node Management EPGs** > **Out-of-Band EPG - default**.

Step 9 In the **Work** pane, expand **Provided Out-of-Band Contracts**.

Step 10 In the **OOB Contract** column, from the drop-down list, choose the out-of-band contract that you created (oob-default). Click **Update**, and click **Submit**.
The contract is associated with the node management EPG.

Step 11 In the **Navigation** pane, right-click **External Network Instance Profile**, and click **Create External Management Entity Instance**.

Step 12 In the **Create External Management Entity Instance** dialog box, perform the following actions:

a) In the **Name** field, enter a name (oob-mgmt-ext).

b) Expand the **Consumed Out-of-Band Contracts** field. From the **Out-of-Band Contract** drop-down list, choose the contract that you created (oob-default). Click **Update**.
Choose the same contract that was provided by the out-of-band management.

c) In the **Subnets** field, enter the subnet address. Click **Submit**.

Only the subnet addresses you choose here will be used to manage the switches. The subnet addresses that are not included cannot be used to manage the switches.

The node management EPG is attached to the external network instance profile. The out-of-band management connectivity is configured.

Configuring Out-of-Band Management Access Using the NX-OS Style CLI

Before You Begin

The APIC out-of-band management connection link must be 1 Gbps.

Procedure

Provide access control for out-of-band management interface to external management subnets as follows:

Example:

```
apic1(config-tenant)# external-l3 epg default oob-mgmt
apic1(config-tenant-l3ext-epg)#match ip 10.0.0.0/8
apic1(config-tenant-l3ext-epg)# exit
apic1(config)# exit
```

Configuring Out-of-Band Management Access Using the REST API

IPv4 and IPv6 addresses are supported for out-of-band management access.

Before You Begin

The APIC out-of-band management connection link must be 1 Gbps.

Procedure

Step 1 Create an out-of-band contract.

Example:

```
POST
https://APIC-IP/api/mo/uni.xml

<polUni>
    <fvTenant name="mgmt">
        <!-- Contract -->
        <vzOOBBrCP name="oob-default">
            <vzSubj name="oob-default">
                <vzRsSubjFiltAtt tnVzFilterName="default" />
            </vzSubj>
        </vzOOBBrCP>
    </fvTenant>
</polUni>
```

Step 2 Associate the out-of-band contract with an out-of-band EPG.

Example:

```
POST
https://APIC-IP/api/mo/uni.xml

<polUni>
    <fvTenant name="mgmt">
        <mgmtMgmtP name="default">
            <mgmtOoB name="default">
                <mgmtRsOoBProv tnVzOOBBrCPName="oob-default" />
            </mgmtOoB>
        </mgmtMgmtP>
    </fvTenant>
</polUni>
```

Step 3 Associate the out-of-band contract with an external management EPG.

Example:

```
POST
https://APIC-IP/api/mo/uni.xml

<polUni>
    <fvTenant name="mgmt">
        <mgmtExtMgmtEntity name="default">
            <mgmtInstP name="oob-mgmt-ext">
                <mgmtRsOoBCons tnVzOOBBrCPName="oob-default" />
                <!-- SUBNET from where switches are managed -->
                <mgmtSubnet ip="10.0.0.0/8" />
            </mgmtInstP>
        </mgmtExtMgmtEntity>
    </fvTenant>
</polUni>
```

Step 4 Create a management address pool.

Example:

```
POST
https://APIC-IP/api/mo/uni.xml

<polUni>
    <fvTenant name="mgmt">
        <fvnsAddrInst name="switchOoboobaddr" addr="172.23.48.1/21">
            <fvnsUcastAddrBlk from="172.23.49.240" to="172.23.49.244"/>
        </fvnsAddrInst>
    </fvTenant>
</polUni>
```

Step 5 Create node management groups.

Example:

```
POST
https://APIC-IP/api/mo/uni.xml

<polUni>
    <infraInfra>
        <infraFuncP>
            <mgmtGrp name="switchOob">
                <mgmtOoBZone name="default">
                    <mgmtRsAddrInst tDn="uni/tn-mgmt/addrinst-switchOoboobaddr" />
                    <mgmtRsOobEpg tDn="uni/tn-mgmt/mgmtp-default/oob-default" />
                </mgmtOoBZone>
            </mgmtGrp>
        </infraFuncP>
        <mgmtNodeGrp name="switchOob">
            <mgmtRsGrp tDn="uni/infra/funcprof/grp-switchOob" />
```

```
            <infraNodeBlk name="default" from_="101" to_="103" />
        </mgmtNodeGrp>
    </infraInfra>
</polUni>
```

Exporting Tech Support, Statistics, and Core Files

About Exporting Files

An administrator can configure export policies in the APIC to export statistics, technical support collections, faults and events, to process core files and debug data from the fabric (the APIC as well as the switch) to any external host. The exports can be in a variety of formats, including XML, JSON, web sockets, secure copy protocol (SCP), or HTTP. You can subscribe to exports in streaming, periodic, or on-demand formats.

An administrator can configure policy details such as the transfer protocol, compression algorithm, and frequency of transfer. Policies can be configured by users who are authenticated using AAA. A security mechanism for the actual transfer is based on a username and password. Internally, a policy element handles the triggering of data.

File Export Guidelines and Restrictions

- HTTP export and the streaming API format is supported only with statistics information. Core and **Tech Support** data are not supported.

Note Do not trigger **Tech Support** from more than five nodes simultaneously, especially if they are to be exported into the APIC or to an external server with insufficient bandwidth and compute resources.

In order to collect **Tech Support** from all the nodes in the fabric periodically, you must create multiple policies. Each policy must cover a subset of the nodes and should be scheduled to trigger in a staggered way (at least 30 minutes apart).

Creating a Remote Location for Exporting Files

This procedure configures the host information and file transfer settings for a remote host that will receive exported files.

Procedure

Step 1 In the menu bar, click **Admin**.

Step 2 In the submenu bar, click **Import/Export**.

Step 3 In the **Navigation** pane, expand **Export Policies**.

Step 4 Right-click **Remote Locations** and choose **Create Remote Path of a File**.

Step 5 In the **Create Remote Path of a File** dialog box, perform the following actions:

a) In the **Name** field, enter a name for the remote location.
b) In the **Host Name/IP** field, enter an IP address or a fully qualified domain name for the destination host.
c) In the **Protocol** field, click the radio button for the desired file transfer protocol.
d) In the **Remote Path** field, type the path where the file will be stored on the remote host.
e) Enter a username and password for logging in to the remote host and confirm the **Password**.
f) From the **Management EPG** drop-down list, choose the management EPG.
g) Click **Submit**.

Sending an On-Demand Techsupport File

Procedure

Step 1 In the menu bar, click **Admin**.

Step 2 In the submenu bar, click **Import/Export**.

Step 3 In the **Navigation** pane, expand **Export Policies**.

Step 4 Right-click **On-demand TechSupport** and choose **Create On-demand TechSupport**.

Step 5 In the **Create On-demand TechSupport** dialog box, perform the following actions:

a) In the **Name** field, enter a name for the techsupport file export policy.
b) To export the file to the controller instead of a remote destination, choose **Export to Controller**.
c) From the **Export Destination** drop-down list, choose the profile of the destination host that will receive the techsupport file.
If no profile appears for the desired destination, you can choose **Create Remote Location** to define it now.
d) From the **Data Container** drop-down list, choose **uni/fabric/tscont**.
e) If the desired source device (leaf or spine) does not appear In the **Source Nodes** table, click the + icon, choose a device, and click **Update**.
f) In the **Source Nodes** table, double-click the source name and click the blue icon to the right of the drop-down list to open the **System Information** window for the source device.
Use the tabs to examine the information of the source device.
g) In the **State** field, click the **triggered** radio button to enable sending of the file.
h) Click **Submit** to send the techsupport file.

Note On-demand tech support files can be saved to another APIC to balance storage and CPU requirements. To verify the location, click on the On-demand TechSupport policy in the **Navigation** pane, then click the **OPERATIONAL** tab in the **Work** pane. The controller is displayed in the **EXPORT LOCATION** field.

i) Right-click the policy name and choose **Collect Tech Support**.

j) Choose **Yes** to begin collecting tech support information.

Overview

This topic provides information on:

- How to use configuration Import and Export to recover configuration states to the last known good state using the Cisco APIC
- How to encrypt secure properties of Cisco APIC configuration files

You can do both scheduled and on-demand backups of user configuration. Recovering configuration states (also known as "roll-back") allows you to go back to a known state that was good before. The option for that is called an Atomic Replace. The configuration import policy (configImportP) supports atomic + replace (importMode=atomic, importType=replace). When set to these values, the imported configuration overwrites the existing configuration, and any existing configuration that is not present in the imported file is deleted. As long as you do periodic configuration backups and exports, or explicitly trigger export with a known good configuration, then you can later restore back to this configuration using the following procedures for the CLI, REST API, and GUI.

For more detailed conceptual information about recovering configuration states using the Cisco APIC, please refer to the *Cisco Application Centric Infrastructure Fundamentals Guide.*

The following section provides conceptual information about encrypting secure properties of configuration files:

Configuration File Encryption

As of release 1.1(2), the secure properties of APIC configuration files can be encrypted by enabling AES-256 encryption. AES encryption is a global configuration option; all secure properties conform to the AES configuration setting. It is not possible to export a subset of the ACI fabric configuration such as a tenant configuration with AES encryption while not encrypting the remainder of the fabric configuration. See the Cisco Application Centric Infrastructure Fundamentals Appendix K: Secure Properties for the list of secure properties.

The APIC uses a 16 to 32 character passphrase to generate the AES-256 keys. The APIC GUI displays a hash of the AES passphrase. This hash can be used to see if the same passphrases was used on two ACI fabrics. This hash can be copied to a client computer where it can be compared to the passphrase hash of another ACI fabric to see if they were generated with the same passphrase. The hash cannot be used to reconstruct the original passphrase or the AES-256 keys.

Observe the following guidelines when working with encrypted configuration files:

- Backward compatibility is supported for importing old ACI configurations into ACI fabrics that use the AES encryption configuration option.

Note Reverse compatibility is not supported; configurations exported from ACI fabrics that have enabled AES encryption cannot be imported into older versions of the APIC software.

- Always enable AES encryption when performing fabric backup configuration exports. Doing so will assure that all the secure properties of the configuration will be successfully imported when restoring the fabric.

Note If a fabric backup configuration is exported without AES encryption enabled, none of the secure properties will be included in the export. Since such an unencrypted backup would not include any of the secure properties, it is possible that importing such a file to restore a system could result in the administrator along with all users of the fabric being locked out of the system.

- The AES passphrase that generates the encryption keys cannot be recovered or read by an ACI administrator or any other user. The AES passphrase is not stored. The APIC uses the AES passphrase to generate the AES keys, then discards the passphrase. The AES keys are not exported. The AES keys cannot be recovered since they are not exported and cannot be retrieved via the REST API.
- The same AES-256 passphrase always generates the same AES-256 keys. Configuration export files can be imported into other ACI fabrics that use the same AES passphrase.
- For troubleshooting purposes, export a configuration file that does not contain the encrypted data of the secure properties. Temporarily turning off encryption before performing the configuration export removes the values of all secure properties from the exported configuration. To import such a configuration file that has all secure properties removed, use the import merge mode; do not use the import replace mode. Using the import merge mode will preserve the existing secure properties in the ACI fabric.
- By default, the APIC rejects configuration imports of files that contain fields that cannot be decrypted. Use caution when turning off this setting. Performing a configuration import inappropriately when this default setting is turned off could result in all the passwords of the ACI fabric to be removed upon the import of a configuration file that does not match the AES encryption settings of the fabric.

Note Failure to observe this guideline could result in all users, including fabric administrations, being locked out of the system.

Creating a Remote Location Using the GUI

This procedure explains how to create a remote location using the APIC GUI.

Procedure

Step 1 On the menu bar, click the **ADMIN** tab.

Step 2 Select **IMPORT/EXPORT**.

Step 3 Under **Import/Export**, click **Remote Locations**.
The **CREATE REMOTE LOCATION** window appears.

Step 4 In the **Description** field, enter a description. (This step is optional.)

Step 5 In the **Host Name (or IP Address)** field, enter an IP address or host name.

Step 6 Specify the protocol by selecting a button for either **scp**, **ftp**, or **sftp**.

Step 7 In the **Remote Path** field, specify a path.

Step 8 In the **Username** field, enter a user name.

Step 9 In the **Password** field, enter a password, then confirm it in the **Confirm Password** field.

Step 10 In the **Management EPG** field, you can specify the inband option or out-of-band option, or you can choose to leave it blank.

Step 11 Click **Submit.**
You have now created a remote location for backing up your data.

Configuring an Export Policy Using the GUI

This procedure explains how to configure an Export policy using the APIC GUI. Follow these steps to trigger a backup of your data:

Procedure

Step 1 On the menu bar, click the **Admin** tab.

Step 2 Select **IMPORT/EXPORT**.

Step 3 Under **Export Policies**, select **Configuration**.

Step 4 Select **Create Configuration Export Policy.**
The **CREATE CONFIGURATION EXPORT POLICY** window appears.

Step 5 In the **Name** field, enter a name for the Export policy.

Step 6 In the **Description** field, enter a description. (This step is optional.)

Step 7 Next to **Format**, select a button for either JSON or XML format.

Step 8 Next to **Start Now**, select a button for either **No** or **Yes** to indicate whether you want to trigger now or trigger based on a schedule. (The easiest method is to choose to trigger immediately.)

Step 9 In the **Target DN** field, enter a name if you want to do a partial backup rather than a backup of the entire configuration. For example, if you only want to back up one specific tenant, you could put in a distinguished name (DN) of the tenant. If you leave it blank, it backs up everything, which is the default.

Step 10 In the **Scheduler** field, select or type to pre-provision.

Step 11 In the **Export Destination** field, specify the remote location where you want to back up the data.

Step 12 Click **Submit**.

You have now created a backup. You can view this under the **Configuration** tab. (The backup file will show in the **Configuration** pane on the right side). There's an Operational tab where you can see if it's running, successful, or failed. If you didn't trigger it yet, it is empty. If you created a backup, it creates a file that is shown in the Operational view of the backup file that was created. If you want to then import that data, you must create an Import policy.

Configuring an Import Policy Using the GUI

This procedure explains how to configure an Import policy using the APIC GUI. Follow these steps to import your backed up data:

Procedure

Step 1 On the menu bar, click the **ADMIN** tab.

Step 2 Select **IMPORT/EXPORT**.

Step 3 Under **Import Policies**, select **Configuration**.

Step 4 Under **Configuration**, select **Create Configuration Import Policy**.
The **CREATE CONFIGURATION IMPORT POLICY** window appears.

Step 5 In the **Name** field, the file name must match whatever was backed up and will have a very specific format. The file name is known to whoever did the backup.

Step 6 The next two options relate to recovering configuration states (also known as "roll-back"). The options are **Input Type** and **Input Mode**. When you recover a configuration state, you want to roll back to a known state that was good before. The option for that is an **Atomic Replace**.
For more detailed information on these input types and modes including **Replace, Merge, Best Effort,** and **Atomic**, refer to the *Cisco Application Centric Infrastructure Fundamentals Guide* .

Step 7 In the **Import Source** field, specify the same remote location that you already created.

Step 8 When you have finished your configuration, click **Start Now**.

Step 9 Click **SUBMIT**.

Configuring an Export Policy Using the NX-OS Style CLI

Before You Begin

If you want to export snapshots according to a schedule, configure a scheduler before configuring the export policy.

Procedure

	Command or Action	Purpose
Step 1	**configure** **Example:** `apic1# configure`	Enters global configuration mode.
Step 2	**[no] snapshot export** *policy-name* **Example:** `apic1(config)# snapshot export myExportPolicy`	Creates a policy for exporting snapshots.
Step 3	**format** {**xml** | **json**} **Example:** `apic1(config-export)# format json`	Specifies the data format for the exported configuration file.
Step 4	**[no] schedule** *schedule-name* **Example:** `apic1(config-export)# schedule EveryEightHours`	(Optional) Specifies an existing scheduler for exporting snapshots.
Step 5	**[no] target** [**infra** | **fabric** | *tenant-name*] **Example:** `apic1(config-export)# target tenantExampleCorp`	(Optional) Assigns the target of the export, which can be fabric, infra, a specific tenant, or none. If no target is specified, all configuration information is exported. The default is no target.
Step 6	**[no] remote path** *remote-path-name* **Example:** `apic1(config-export)# remote path myBackupServer`	(Optional) Specifies the name of a configured remote path to which the file will be sent. If no remote path is specified, the file is exported locally to a folder in the controller. The default is no remote path.
Step 7	**end** **Example:** `apic1(config-export)# end`	Returns to EXEC mode.
Step 8	**trigger snapshot export** *policy-name* **Example:** `apic1# trigger snapshot export myExportPolicy`	Executes the snapshot export task. If the export policy is configured with a scheduler, this step is unnecessary unless you want an immediate export.

Examples

This example shows how to configure the periodic export of a JSON-format snapshot file for a specific tenant configuration.

```
apic1# configure
apic1(config)# snapshot export myExportPolicy
apic1(config-export)# format json
apic1(config-export)# target tenantExampleCorp
apic1(config-export)# schedule EveryEightHours
```

Configuring an Import Policy Using the NX-OS Style CLI

To configure an import policy using the NX-OS Style CLI, enter the following:

Procedure

	Command or Action	Purpose
Step 1	**configure** **Example:** `apic1# configure`	Enters global configuration mode.
Step 2	**[no] snapshot import** *policy-name* **Example:** `apic1(config)# snapshot import myImportPolicy`	Creates a policy for importing snapshots.
Step 3	**file** *filename* **Example:** `apic1(config-import)# file ce2_DailyAutoBackup-2015-11-21T01-00-17.tar.gz`	Specifies the name of the file to be imported.
Step 4	**action** {**merge** | **replace**} **Example:** `apic1(config-import)# action replace`	Specifies whether the imported configuration settings will be merged with the current settings or whether the imported configuration will completely replace the current configuration.
Step 5	**[no] mode** {**atomic** | **best-effort**} **Example:** `apic1(config-import)# mode atomic`	Specifies how the import process handles configuration errors when applying the imported settings. The best-effort import mode allows skipping individual configuration errors in the archive, while atomic mode cancels the import upon any configuration error.
Step 6	**[no] remote path** *remote-path-name* **Example:** `apic1(config-import)# remote path myBackupServer`	(Optional) Specifies the name of a configured remote path from which the file will be imported. If no remote path is specified, the file is

	Command or Action	Purpose
		imported locally from a folder in the controller. The default is no remote path.
Step 7	**end** **Example:** `apic1(config-import)# end`	Returns to EXEC mode.
Step 8	**trigger snapshot import** *policy-name* **Example:** `apic1# trigger snapshot import myImportPolicy`	Executes the snapshot import task.

Examples

This example shows how to configure and execute the importing of a snapshot file to replace the current configuration.

```
apic1# show snapshot files
File    : ce2_DailyAutoBackup-2015-11-21T01-00-17.tar.gz
Created : 2015-11-21T01:00:21.167+00:00
Root    :
Size    : 22926

apic1# configure
apic1(config)# snapshot import myImportPolicy
apic1(config-import)# file ce2_DailyAutoBackup-2015-11-21T01-00-17.tar.gz
apic1(config-import)# action replace
apic1(config-import)# mode atomic
apic1(config-import)# end
apic1# trigger snapshot import myImportPolicy
```

Configuring an Export Policy Using the REST API

To configure an export policy using the REST API:

```
POST
https://<ip-of-apic>/api/mo/uni/fabric.xml
<fabricInst dn="uni/fabric">
<configExportP name="export" format="xml" adminSt="triggered">
<configRsExportDestination tnFileRemotePathName="backup" />
</configExportP>
<fileRemotePath name="backup" host="10.10.10.1" protocol="scp"
remotePath="/home/user" userName="user" userPasswd="pass" />
</fabricInst>
```

Configuring an Import Policy Using the REST API

To configure an import policy using the REST API:

```
POST
https://<ip-of-apic>/api/mo/uni/fabric.xml
```

```
<fabricInst dn="uni/fabric">
<configImportP name="imp" fileName="aa.tar.gz" adminSt="triggered" importType="replace"
importMode="best-effort">
<configRsImportSource tnFileRemotePathName="backup" />
</configImportP>
<fileRemotePath name="backup" host="10.10.10.1" protocol="scp"
remotePath="/home/user" userName="user" userPasswd="pass" />
</fabricInst>
```

Encrypting Configuration Files Using the GUI

To encrypt a configuration file using the APIC GUI:

Procedure

Step 1 On the menu bar, select the **ADMIN** tab.

Step 2 Select the **AAA** tab under the **ADMIN** tab.

Step 3 Select **AES Encryption Passphrase and Keys for Config Export (and Import)** from the left navigation pane.
The **Global AES Encryption Settings for all Configurations Import and Export** window displays in the right pane.

Step 4 Create a passphrase, which can be between 16 and 32 characters long. There are no restrictions on the type of characters used.

Step 5 Click **SUBMIT**.

Note Once you have created and posted the passphrase, the keys are then generated in the back-end and the passphrase is not recoverable. Therefore, your passphrase is not visible to anyone because the key is automatically generated then deleted. Your backup only works if you know the passphrase (no one else can open it).

The **Key Configured** field now shows **yes**. You now see an encrypted hash (which is not the actual passphrase, but just a hash of it) in the **Encrpyted Passphrase** field.

Step 6 Once you have set and confirmed your passphrase, click the checkbox next to **Enable Encryption** to turn the AES encryption feature on or off.

Note When this box is unchecked (off) and encryption is disabled, all exported configurations (exports) are missing the secure fields (such as passwords and certificates). When this box is checked (on), all exports show the secure fields.

Step 7 Select the **IMPORT/EXPORT** tab under the **ADMIN** tab.

Step 8 Select **Import Policies** from the left navigation pane.

Step 9 Select **Configuration** under **Import Policies**.
If you previously turned **Enable Encryption** on, there is a configuration import policy (or list of policies) shown in the left navigation pane under **Configuration** that you can set properties for.

Step 10 Ensure that the checkbox next to **Fail Import if secure fields cannot be decrypted** is checked (which is the default selection).

Note This checkbox is enabled by default. It is highly recommended that you do not uncheck this box when you import the configuration. If you uncheck this box, the system attempts to import all the fields, however, any fields that it cannot encrypt are blank/missing. As a result, you could lock yourself out of the system because the admin passwords could go blank/missing (if you lock yourself out of the system, refer to Cisco APIC Troubleshooting Guide). Unchecking the box launches a warning message pop-up screen. If the box is checked, there are security checks that prevent lockouts and the configuration does not import.

Step 11 You can also set properties for exporting configuration files in the **Configuration** tab under the **Export Policies** tab in the left navigation pane.

Follow the same steps as previously described for setting properties for configuration import policies.

Note You cannot configure a passphrase in this section. The one you previously set is now global across all configurations in this box and across all tenants. If you export a configuration from this tab (you have configured a passphrase and enabled encryption) you get a complete backup file. If encryption is not enabled, you get a backup file with the secure properties removed. These backup files are useful when exporting to TAC support engineers, for example, because all the secure fields are missing. This is true for any secure properties in the configuration. There is also a clear option that clears the encryption key.

Step 12 Note the list of the configuration import behaviors and associated results in the following table:

Configuration Import Behavior Scenario	Result
Old configuration from previous release	Import of configurations from old releases is fully supported and successfully imports all secure fields stored in old configurations.
Configuration import when AES encryption is not configured	If the import is for a configuration without secure fields, it is successful with the behavior previously described. If the imported configuration has secure fields, it is rejected.
Configuration import when AES passphrases do not match	If the import is for a configuration without secure fields, it is successful with the behavior previously described. If the imported configuration has secure fields, it is rejected.
Configuration import when AES passphrases match	Import is successful
Configuration import when AES passphrases do not match for copy/pasted fields	This specific case occurs when the user has copied and pasted secure fields from other configurations that were exported with a different passphrase. During the first pass parsing of the imported backup file, if any property fails to decrypt correctly, the import fails without importing any shards. Therefore, if a shard fails to decrypt all properties, all shards are rejected.

Encrypting Configuration Files Using the NX-OS Style CLI

To encrypt a configuration file using the NX-OS Style CLI:

```
apic1# configure
apic1(config)# crypto aes
 <CR>
apic1(config)# crypto aes
apic1(config-aes)#
 clear-encryption-key  Clears AES encryption key
encryption            Enable AES Encryption
no                    Negate a command or set its defaults
passphrase            Configure passphrase for AES encryption

bash                  bash shell for unix commands
end                   Exit to the exec mode
exit                  Exit from current mode
fabric                show fabric related information
show                  Show running system information
where                 show the current mode
apic1(config-aes)# encryption
 <CR>
apic1(config-aes)# encryption
apic1(config-aes)#
 clear-encryption-key  Clears AES encryption key
encryption            Enable AES Encryption
no                    Negate a command or set its defaults
passphrase            Configure passphrase for AES encryption

bash                  bash shell for unix commands
end                   Exit to the exec mode
exit                  Exit from current mode
fabric                show fabric related information
show                  Show running system information
where                 show the current mode
apic1(config-aes)# passphrase
 WORD  Passphrase for AES encryption (Range of chars: 16-32) in quotes
apic1(config-aes)# passphrase "abcdefghijklmnopqrstuvwxyz"
apic1(config-aes)#
```

Encrypting Configuration Files Using the REST API

To encrypt a configuration file using the REST API, enter the following:

```
POST
https://<ip-of-apic>/api/mo/uni/fabric.xml
<pkiExportEncryptionKey passphrase="abcdefghijklmnopqrstuvwxyz"
strongEncryptionEnabled="true"/>
```

Backing up, Restoring, and Rolling Back Controller Configuration

This section describes the set of features for backing up (creating snapshots), restoring, and rolling back a controller configuration.

Workflow

This section describes the workflow of the features for backing up, restoring, and rolling back configuration files. All of the features described in this document follow the same workflow pattern. Once the corresponding policy is configured, **admintSt** must be set to **triggered** in order to trigger the job.

Once triggered, an object of type **configJob** (representing that run) is created under a container object of type **configJobCont** (the naming property value is set to the policy DN). The container's **lastJobName** field can be used to determine the last job that was triggered for that policy.

Note Up to five configJob objects are kept under a single job container at a time, with each new job triggered. The oldest job is removed to ensure this.

The **configJob** object contains the following information:

- execution time
- name of the file being processed/generated
- status, as follows:
  - pending
  - running
  - failed
  - fail-no-data
  - success
  - success-with-warnings
- details string (failure messages and warnings)
- progress percentage = 100 * lastStepIndex/totalStepCount
- lastStepDescr field indicating what was being done last

Remote Path

The fileRemotePath object holds the following remote location path parameters:

- hostname or IP

- port
- protocol: ftp, scp, and others
- remote directory (not file path)
- username
- password

Note The password must be re-submitted every time changes are made.

Sample Configuration

The following is a sample configuration:

Under **fabricInst** (uni/fabric), enter:

```
<fileRemotePath name="path-name" host="host name or ip" protocol="scp"
remotePath="path/to/some/folder" userName="user-name" userpasswd="password" />
```

Configuration Export to Controller

The configuration export extracts user-configurable managed object (MO) trees from all thirty-two shards in the cluster, writes them into separate files, then compresses them into a tar gzip. The configuration export then uploads the tar gzip to a pre-configured remote location (configured via **configRsRemotePath** pointing to a **fileRemotePath** object) or stores it as a **snapshot** on the controller(s).

Note See the Snapshots section for more details.

The **configExportP** policy is configured as follows:

- **name** - policy name
- **format** - format in which the data is stored inside the exported archive (xml or json)
- **targetDn** - the domain name (DN) of the specific object you want to export (empty means everything)
- **snapshot** - when true, the file is stored on the controller, no remote location configuration is needed
- **includeSecureFields** - Set to true by default, indicates whether the encrypted fields (passwords, etc.) should be included in the export archive.

Note The **configSnapshot** object is created holding the information about this snapshot (see the Snapshots section).

Scheduling Exports

An export policy can be linked with a scheduler, which triggers the export automatically based on a pre-configured schedule. This is done via the **configRsExportScheduler** relation from the policy to a **trigSchedP** object (see the following Sample Configuration section).

Note A scheduler is optional. A policy can be triggered at any time by setting the adminSt to **triggered**.

Troubleshooting

If you get an error message indicating that the generated archive could not be uploaded to the remote location, refer to the Connectivity Issues section.

Sample Configuration Using the NX-OS Style CLI

The following is a sample configuration using the NX-OS Style CLI:

```
apic1(config)# snapshot
 download  Configuration snapshot download setup mode
export    Configuration export setup mode
import    Configuration import setup mode
rollback  Configuration rollback setup mode
upload    Configuration snapshot upload setup mode
apic1(config)# snapshot export policy-name
apic1(config-export)#
 format    Snapshot format: xml or json
no        Negate a command or set its defaults
remote    Set the remote path configuration will get exported to
schedule  Schedule snapshot export
target    Snapshot target

bash      bash shell for unix commands
end       Exit to the exec mode
exit      Exit from current mode
fabric    show fabric related information
show      Show running system information
where     show the current mode
apic1(config-export)# format xml
apic1(config-export)# no remote path             [If no remote path is specified, the file
 is exported locally to a folder in the controller]
apic1(config-export)# target                     [Assigns the target of the export, which
can be fabric, infra, a specific tenant, or none. If no target is specified, all configuration
 information is exported.]
WORD  infra, fabric or tenant-x
apic1(config-export)#
apic1# trigger snapshot export policy-name       [Executes the snapshot export task]
```

Sample Configuration Using the GUI

The following is a sample configuration using the GUI:

1. On the menu bar, click the **ADMIN** tab.
2. Select **IMPORT/EXPORT**.
3. Under **Export Policies**, select **Configuration**.
4. Under Configuration, click the configuration that you would like to roll back to. For example, you can click **defaultOneTime**, which is the default.
5. Next to **Format**, select a button for either JSON or XML format.
6. Next to **Start Now**, select a button for either **No** or **Yes** to indicate whether you want to trigger now or trigger based on a schedule. (The easiest method is to choose to trigger immediately.)
7. For the **Target DN** field, enter the name of the tenant configuration you are exporting.

8 If you want to store the configuration on the controller itself, check the **Snapshot** option. If you want to configure a remote location, uncheck this option.
9 For the **Scheduler** field, you have the option to create a scheduler instructing when and how often to export the configuration.
10 For the **Encryption** field, you have the option to enable or disable the encryption of your configuration file.
11 When you have finished your configuration, click **Start Now**.
12 Click **SUBMIT** to trigger your configuration export.

Sample Configuration Using REST API

The following is a sample configuration using the REST API:

```
<configExportP name="policy-name" format="xml" targetDn="/some/dn or empty which means
everything"
snapshot="false" adminSt="triggered">
<configRsRemotePath tnFileRemotePathName="some remote path name" />
<configRsExportScheduler tnTrigSchedPName="some scheduler name" />
</configExportP>
```

Note When providing a remote location, if you set the snapshot to True, the backup ignores the remote path and stores the file on the controller.

Configuration Import to Controller

Configuration import downloads, extracts, parses, analyzes and applies the specified, previously exported archive one shard at a time in the following order: infra, fabric, tn-common, then everything else. The fileRemotePath configuration is performed the same way as for export (via configRsRemotePath). Importing snapshots is also supported.

The **configImportP** policy is configured as follows:

- **name** - policy name
- **fileName** - name of the archive file (not the path file) to be imported
- **importMode**
  - Best-effort mode: each MO is applied individually, and errors only cause the invalid MOs to be skipped.

    Note If the object is not present on the controller, none of the children of the object get configured. Best-effort mode attempts to configure the children of the object.
  - Atomic mode: configuration is applied by whole shards. A single error causes whole shard to be rolled back to its original state.
- **importType**
  - replace - Current system configuration is replaced with the contents or the archive being imported (only atomic mode is supported)

    - merge - Nothing is deleted, archive content is applied on top the existing system configuration.

- **snapshot** - when true, the file is taken from the controller and no remote location configuration is needed.
- **failOnDecryptErrors** - (true by default) the file fails to import if the archive was encrypted with a different key than the one that is currently set up in the system.

Troubleshooting

The following scenarios may need troubleshooting:

- If the generated archive could not be downloaded from the remote location, refer to the Connectivity Issues section.
- If the import succeeded with warnings, check the details.
- If a file could not be parsed, refer to the following scenarios:
    - If the file is not a valid XML or JSON file, check whether or not the files from the exported archive were manually modified.
    - If an object property has an unknown property or property value, it may be because:
        - The property was removed or an unknown property value was manually entered
        - The model type range was modified (non-backward compatible model change)
        - The naming property list was modified
- If an MO could not be configured, note the following:
    - Best-effort mode logs the error and skips the MO
    - Atomic mode logs the error and skips the shard

Sample Configuration Using the NX-OS Style CLI

The following is a sample configuration using the NX-OS Style CLI:

```
apic1# configure
apic1(config)# snapshot
 download  Configuration snapshot download setup mode
export    Configuration export setup mode
import    Configuration import setup mode
rollback  Configuration rollback setup mode
upload    Configuration snapshot upload setup mode
apic1(config)# snapshot import
 WORD       Import configuration name
default
rest-user
apic1(config)# snapshot import policy-name
apic1(config-import)#
 action  Snapshot import action merge|replace
file    Snapshot file name
mode    Snapshot import mode atomic|best-effort
no      Negate a command or set its defaults
remote  Set the remote path configuration will get imported from

bash    bash shell for unix commands
end     Exit to the exec mode
exit    Exit from current mode
```

```
fabric  show fabric related information
show    Show running system information
where   show the current mode
apic1(config-import)# file < from "show snapshot files" >
apic1(config-import)# no remote path
apic1(config-import)#
apic1# trigger snapshot import policy-name       [Executes the snapshot import task]
```

Sample Configuration Using the GUI

The following is a sample configuration using the GUI:

1. On the menu bar, click the **ADMIN** tab.
2. Select **IMPORT/EXPORT**.
3. Under **Import Policies**, select **Configuration**.
4. Under **Configuration**, select **Create Configuration Import Policy**. The **CREATE CONFIGURATION IMPORT POLICY** window appears.
5. In the **Name** field, the file name must match whatever was backed up and will have a very specific format. The file name is known to whoever did the backup.
6. The next two options relate to recovering configuration states (also known as "roll-back"). The options are **Input Type** and **Input Mode**. When you recover a configuration state, you want to roll back to a known state that was good before. The option for that is an **Atomic Replace**.
7. If you want to store the configuration on the controller itself, check the **Snapshot** option. If you want to configure a remote location, uncheck this option.
8. In the **Import Source** field, specify the same remote location that you already created.
9. For the **Encryption** field, you have the option to enable or disable the encryption of your configuration file.
10. Click **SUBMIT** to trigger your configuration import.

Sample Configuration Using the REST API

The following shows a sample configuration using the REST API:

```
<configImportP name="policy-name" fileName="someexportfile.tgz" importMode="atomic"
importType="replace" snapshot="false" adminSt="triggered">
<configRsRemotePath tnFileRemotePathName="some remote path name" />
</configImportP>
```

Snapshots

Snapshots are configuration backup archives, stored (and replicated) in a controller managed folder. To create one, an export can be performed with the **snapshot** property set to true. In this case, no remote path configuration is needed. An object of **configSnapshot** type is created to expose the snapshot to the user.

configSnapshot objects provide the following:

- file name
- file size
- creation date

- root DN indicating what the snapshot is of (fabric, infra, specific tenant, and so on)
- ability to remove a snapshot (by setting the retire field to true)

To import a snapshot, set the import policy snapshot property to true and provide the name of the snapshot file (from configSnapshot).

Snapshot Manager Policy

The **configSnapshotManagerP** policy allows you to create snapshots from remotely stored export archives. You can attach a remote path to the policy, provide the file name (same as with configImportP), set the mode to download, and trigger. The manager downloads the file, analyzes it to make sure the archive is valid, stores it on the controller, and creates the corresponding configSnapshot object. The snapshot manager also allow you to upload a snapshot archive to a remote location. In this case, the mode must be set to upload.

Troubleshooting

For troubleshooting, refer to the Connectivity Issues section.

Snapshot Upload from Controller to Remote Path Using the NX-OS CLI

```
apic1(config)# snapshot upload policy-name
apic1(config-upload)#
 file    Snapshot file name
no      Negate a command or set its defaults
remote  Set the remote path configuration will get uploaded to

bash    bash shell for unix commands
end     Exit to the exec mode
exit    Exit from current mode
fabric  show fabric related information
show    Show running system information
where   show the current mode
apic1(config-upload)# file <file name from "show snapshot files">
apic1(config-upload)# remote path remote-path-name
apic1# trigger snapshot upload policy-name       [Executes the snapshot upload task]
```

Snapshot Download from Controller to Remote Path Using the NX-OS CLI

```
apic1(config)# snapshot download policy-name
apic1(config-download)#
 file    Snapshot file name
no      Negate a command or set its defaults
remote  Set the remote path configuration will get downloaded from

bash    bash shell for unix commands
end     Exit to the exec mode
exit    Exit from current mode
fabric  show fabric related information
show    Show running system information
where   show the current mode
apic1(config-download)# file < file from remote path>
apic1(config-download)# remote path remote-path-name
apic1# trigger snapshot download policy-name       [Executes the snapshot download task]
```

Snapshot Upload and Download Using the GUI

To upload a snapshot file to a remote location:

1 Right-click on the snapshot file listed in the **Config Rollbacks** pane, and select the **Upload to Remote Location option**. The **Upload snapshot to remote location** box appears.

2 Click **SUBMIT**.

To download a snapshot file from a remote location:

1 Click the import icon on the upper right side of the screen. The **Import remotely stored export archive to snapshot** box appears.
2 Enter the file name in the **File Name** field.
3 Select a remote location from the Import Source pull-down, or check the box next to **Or create a new one** to create a new remote location.
4 Click **SUBMIT**.

Snapshot Upload and Download Using the REST API

```
<configSnapshotManagerP name="policy-name" fileName="someexportfile.tgz"
mode="upload|download" adminSt="triggered">
<configRsRemotePath tnFileRemotePathName="some remote path name" />
</configSnapshotManagerP>
```

Rollback

The **configRollbackP** policy is used to undo the changes made between two snapshots. Objects are processed as follows:

- Deleted MOs are recreated
- Created MOs are deleted
- Modified MOs are reverted

Note The rollback feature only operates on snapshots. Remote archives are not supported. To use one, the snapshot manager can be used to create a snapshot from it for the rollback. The policy does not require a remote path configuration. If one is provided, it will be ignored.

Rollback Workflow

The policy snapshotOneDN and snapshotTwoDn fields must be set and the first snapshot (S1) must precede snapshot two (S2). Once triggered, snapshots are extracted and analyzed, and the difference between them is calculated and applied.

MOs are located that are:

- Present in S1 but not present in S2 - these MOs are deleted and rollback re-creates them
- Not present in S1 but not present in S2 - these MOs are created after S1 and rollback deletes them if:
  - These MOs are not modified after S2 is taken
  - None of the MO's descendants are created or modified after S2 is taken
- Present in both S1 and S2, but with different property values - these MO properties are reverted to S1, unless the property was modified to a diffferent value after S2 is taken. In this case, it is left as is.

The rollback feature also generates a diff file that contains the confiuration generated as a result of these calculations. Applying this configuration is the last step of the rollback process. The content of this file can be retrieved via a special REST API called readiff: apichost/mqapi2/snapshots.readiff.xml?jobdn=SNAPSHOT_JOB_DN.

Rollback (which is difficult to predict) also has a preview mode (set preview to true), which prevents rollback from making any actual changes. It calculates and generates the diff file, allowing you to preview what exactly is going to happen once the rollback is actually performed.

Diff Tool

Another special REST API is available, which provides diff functionality between two snapshots: apichost/mqapi2/snapshots.diff.xml?s1dn=SNAPSHOT_ONE_DN&s2dn=SNAPSHOT_TWO_DN.

Sample Configuration Using the NX-OS Style CLI

This example shows how to configure and execute a rollback using the NX-OS Style CLI:

```
apic1# show snapshot files
File    : ce2_DailyAutoBackup-2015-11-21T01-00-17.tar.gz
Created : 2015-11-21T01:00:21.167+00:00
Root    :
Size    : 22926

File    : ce2_DailyAutoBackup-2015-11-21T09-00-21.tar.gz
Created : 2015-11-21T09:00:24.025+00:00
Root    :
Size    : 23588

apic1# configure
apic1(config)# snapshot rollback myRollbackPolicy
apic1(config-rollback)# first-file ce2_DailyAutoBackup-2015-11-21T01-00-17.tar.gz
apic1(config-rollback)# second-file ce2_DailyAutoBackup-2015-11-21T09-00-21.tar.gz
apic1(config-rollback)# preview
apic1(config-rollback)# end
apic1# trigger snapshot rollback myRollbackPolicy
```

Sample Configuration Using the GUI

This example shows how to configure and execute a rollback using the GUI:

1. On the menu bar, click the **Admin** tab.
2. Click **Config Rollbacks**, located under the Admin tab.
3. Select the first configuration file from the **Config Rollbacks** list (in the left-side pane).
4. Select the second configuration file in the **Configuration for selected snapshot** pane (in the right-side pane).
5. Click the **Compare with previous snapshot** drop-down menu (at the bottom of the right-side pane), then select the second configuration file from that list. A diff file is then generated so that you can compare the differences between the two snapshots.

Note After the file generates, there is an option to undo these changes.

Sample Configuration Using the REST API

This example shows how to configure and execute a rollback using the REST API:

```
<configRollbackP name="policy-name" snapshotOneDn="dn/of/snapshot/one"
snapshotOneDn="dn/of/snapshot/two" preview="false" adminSt="triggered" />
```

Using Syslog

About Syslog

During operation, a fault or event in the Cisco Application Centric Infrastructure (ACI) system can trigger the sending of a system log (syslog) message to the console, to a local file, and to a logging server on another system. A system log message typically contains a subset of information about the fault or event. A system log message can also contain audit log and session log entries.

Note For a list of syslog messages that the APIC and the fabric nodes can generate, see http://www.cisco.com/c/en/us/td/docs/switches/datacenter/aci/apic/sw/1-x/syslog/guide/aci_syslog/ACI_SysMsg.html.

Many system log messages are specific to the action that a user is performing or the object that a user is configuring or administering. These messages can be the following:

- Informational messages, providing assistance and tips about the action being performed
- Warning messages, providing information about system errors related to an object, such as a user account or service profile, that the user is configuring or administering

In order to receive and monitor system log messages, you must specify a syslog destination, which can be the console, a local file, or one or more remote hosts running a syslog server. In addition, you can specify the minimum severity level of messages to be displayed on the console or captured by the file or host. The local file for receiving syslog messages is `/var/log/external/messages`.

A syslog source can be any object for which an object monitoring policy can be applied. You can specify the minimum severity level of messages to be sent, the items to be included in the syslog messages, and the syslog destination.

Additional details about the faults or events that generate these system messages are described in the *Cisco APIC Faults, Events, and System Messages Management Guide*, and system log messages are listed in the *Cisco ACI System Messages Reference Guide*.

Note Not all system log messages indicate problems with your system. Some messages are purely informational, while others may help diagnose problems with communications lines, internal hardware, or the system software.

Creating a Syslog Destination and Destination Group

This procedure configures syslog data destinations for logging and evaluation. You can export syslog data to the console, to a local file, or to one or more syslog servers in a destination group.

Procedure

Step 1 In the menu bar, click **Admin**.

Step 2 In the submenu bar, click **External Data Collectors**.

Step 3 In the **Navigation** pane, expand **Monitoring Destinations**.

Step 4 Right-click **Syslog** and choose **Create Syslog Monitoring Destination Group**.

Step 5 In the **Create Syslog Monitoring Destination Group** dialog box, perform the following actions:

a) In the group and profile **Name** field, enter a name for the monitoring destination group and profile.

b) In the group and profile **Admin State** drop-down list, choose **enabled**.

c) To enable sending of syslog messages to a local file, choose **enabled** from the Local File Destination **Admin State** drop-down list and choose a minimum severity from the Local File Destination **Severity** drop-down list.
The local file for receiving syslog messages is `/var/log/external/messages`.

d) To enable sending of syslog messages to the console, choose **enabled** from the Console Destination **Admin State** drop-down list and choose a minimum severity from the Console Destination **Severity** drop-down list.

e) Click **Next**.

f) In the **Create Remote Destinations** area, click + to add a remote destination.

Step 6 In the **Create Syslog Remote Destination** dialog box, perform the following actions:

a) In the **Host** field, enter an IP address or a fully qualified domain name for the destination host.

b) (Optional) In the **Name** field, enter a name for the destination host.

c) In the **Admin State** field, click the **enabled** radio button.

d) (Optional) Choose a minimum severity **Severity**, a **Port** number, and a syslog **Forwarding Facility**.

e) From the **Management EPG** drop-down list, choose the management endpoint group.

f) Click **OK**.

Step 7 (Optional) To add more remote destinations to the remote destination group, click + again and repeat the steps in the **Create Syslog Remote Destination** dialog box

Step 8 Click **Finish**.

Creating a Syslog Source

A syslog source can be any object for which an object monitoring policy can be applied.

Before You Begin

Create a syslog monitoring destination group.

Procedure

Step 1 From the menu bar and the navigation frame, navigate to a **Monitoring Policies** menu for the area of interest. You can configure monitoring policies for tenants, fabric, and access.

Step 2 Expand **Monitoring Policies**, then select and expand a monitoring policy.
Under **Fabric > Fabric Policies > Monitoring Policies > Common Policy** is a basic monitoring policy that applies to all faults and events and is automatically deployed to all nodes and controllers in the fabric. Alternatively, you can specify an existing policy with a more limited scope.

Step 3 Under the monitoring policy, click **Callhome/SNMP/Syslog**.

Step 4 In the **Work** pane, choose **Syslog** from the **Source Type** drop-down list.

Step 5 From the **Monitoring Object** list, choose a managed object to be monitored.
If the desired object does not appear in the list, follow these steps:

a) Click the Edit icon to the right of the **Monitoring Object** drop-down list.
b) From the **Select Monitoring Package** drop-down list, choose an object class package.
c) Select the checkbox for each object that you want to monitor.
d) Click **Submit**.

Step 6 In a tenant monitoring policy, if you select a specific object instead of **All**, a **Scope** selection appears.
In the **Scope** field, select a radio button to specify the system log messages to send for this object:

- **all—**Send all events and faults related to this object
- **specific event—**Send only the specified event related to this object. From the **Event** drop-down list, choose the event policy.
- **specific fault—**Send only the specified fault related to this object. From the **Fault** drop-down list, choose the fault policy.

Step 7 Click + to create a syslog source.

Step 8 In the **Create Syslog Source** dialog box, perform the following actions:

a) In the **Name** field, enter a name for the syslog source.
b) From the **Min Severity** drop-down list, choose the minimum severity of system log messages to be sent.
c) In the **Include** field, check the checkboxes for the type of messages to be sent.
d) From the **Dest Group** drop-down list, choose the syslog destination group to which the system log messages will be sent.
e) Click **Submit**.

Step 9 (Optional) To add more syslog sources, click + again and repeat the steps in the **Create Syslog Source** dialog box

Out-of-Band DNS Connection

Note Some applications such as Tech Support and Cisco Call Home require an in-band and out-of-band DNS connection on the leaf switches to successfully resolve host names.

Using Atomic Counters

About Atomic Counters

Atomic counters allow you to gather statistics about traffic between flows. Using atomic counters, you can detect drops and misrouting in the fabric, enabling quick debugging and isolation of application connectivity issues. For example, an administrator can enable atomic counters on all leaf switches to trace packets from endpoint 1 to endpoint 2. If any leaf switches have nonzero counters, other than the source and destination leaf switches, an administrator can drill down to those leafs.

In conventional settings, it is nearly impossible to monitor the amount of traffic from a bare metal NIC to a specific IP address (an endpoint) or to any IP address. Atomic counters allow an administrator to count the number of packets that are received from a bare metal endpoint without any interference to its data path. In addition, atomic counters can monitor per-protocol traffic that is sent to and from an endpoint or an application group.

Leaf-to-leaf (TEP-to-TEP) atomic counters can provide the following:

- Counts of sent, received, dropped, and excess packets
  - Sent packets: The sent number reflects how many packets were sent from the source TEP (tunnel endpoint) to the destination TEP.
  - Received packets: The received number reflects how many packets the destination TEP received from the source TEP.
  - Dropped packets: The dropped number reflects how many packets were dropped during transmission. This number is the difference in the amount of packets sent and the amount of packets received.
  - Excess packets: The excess number reflects how many extra packets were received during transmission. This number is the amount of packets that were unexpectedly received due to a forwarding mismatch or a misrouting to the wrong place.
- Short-term data collection such as the last 30 seconds, and long-term data collection such as 5 minutes, 15 minutes, or more
- A breakdown of per-spine traffic
- Ongoing monitoring

Note Leaf-to-leaf (TEP to TEP) atomic counters are cumulative and cannot be cleared. However, because 30-second atomic counters reset at 30-second intervals, they can be used to isolate intermittent or recurring problems. Atomic counters require an active fabric Network Time Protocol (NTP) policy.

Tenant atomic counters can provide the following:

- Application-specific counters for traffic across the fabric, including sent, received, dropped, and excess packets
- Modes include the following:
  - EPtoEP (endpoint to endpoint)

- EPGtoEPG (endpoint group to endpoint group)

Note For EPGtoEPG, the options include ipv4 only, ipv6 only, and ipv4, ipv6. Any time there is an ipv6 option, you use twice the TCAM entries, which means the scale numbers may be less than expected for pure ipv4 policies.

- EPGtoEP (endpoint group to endpoint)
- EPtoAny (endpoint to any)
- AnytoEP (any to endpoint)
- EPGtoIP (endpoint group to IP, used only for external IP address)
- EPtoExternalIP (endpoint to external IP address)

Atomic Counters Guidelines and Restrictions

- Use of atomic counters is not supported when the endpoints are in different tenants or in different contexts (VRFs) within the same tenant.
- In pure layer 2 configurations where the IP address is not learned (the IP address is 0.0.0.0), endpoint-to-EPG and EPG-to-endpoint atomic counter policies are not supported. In these cases, endpoint-to-endpoint and EPG-to-EPG policies are supported. External policies are virtual routing and forwarding (VRF)-based, requiring learned IP addresses, and are supported.
- When the atomic counter source or destination is an endpoint, the endpoint must be dynamic and not static. Unlike a dynamic endpoint (fv:CEp), a static endpoint (fv:StCEp) does not have a child object (fv:RsCEpToPathEp) that is required by the atomic counter.
- In a transit topology, where leaf switches are not in full mesh with all spine switches, then leaf-to-leaf (TEP to TEP) counters do not work as expected.
- For leaf-to-leaf (TEP to TEP) atomic counters, once the number of tunnels increases the hardware limit, the system changes the mode from trail mode to path mode and the user is no longer presented with per-spine traffic.
- The atomic counter does not count spine proxy traffic.
- Packets dropped before entering the fabric or before being forwarded to a leaf port are ignored by atomic counters.
- Packets that are switched in the hypervisor (same Port Group and Host) are not counted.
- Atomic counters require an active fabric Network Time Protocol (NTP) policy.
- An atomic counter policy configured with fvCEp as the source and/or destination counts only the traffic that is from/to the MAC and IP addresses that are present in the fvCEp managed objects (MOs). If the fvCEp MO has an empty IP address field, then all traffic to/from that MAC address would be counted regardless of the IP address. If the APIC has learned multiple IP addresses for an fvCEp, then traffic from only the one IP address in the fvCEp MO itself is counted as previously stated. In order to configure an atomic counter policy to/from a specific IP address, use the fvIp MO as the source and/or destination.
- If there is an fvIp behind an fvCEp, you must add fvIP-based policies and not fvCEp-based policies.

Configuring Atomic Counters

Procedure

Step 1 In the menu bar, click **Tenants**.

Step 2 In the submenu bar, click the desired tenant.

Step 3 In the **Navigation** pane, expand the tenant and expand **Troubleshoot Policies**.

Step 4 Under **Troubleshoot Policies**, expand **Atomic Counter Policy** and choose a traffic topology.
You can measure traffic between a combination of endpoints, endpoint groups, external interfaces, and IP addresses.

Step 5 Right-click the desired topology and choose **Add** *topology* **Policy** to open an **Add Policy** dialog box.

Step 6 In the **Add Policy** dialog box, perform the following actions:

a) In the **Name** field, enter a name for the policy.

b) choose or enter the identifying information for the traffic source.
The required identifying information differs depending on the type of source (endpoint, endpoint group, external interface, or IP address).

c) choose or enter the identifying information for the traffic destination.

d) (Optional) (Optional) In the **Filters** table, click the + icon to specify filtering of the traffic to be counted. In the resulting **Create Atomic Counter Filter** dialog box, you can specify filtering by the IP protocol number (TCP=6, for example) and by source and destination IP port numbers.

e) Click **Submit** to save the atomic counter policy.

Step 7 In the **Navigation** pane, under the selected topology, choose the new atomic counter policy.
The policy configuration is displayed in the **Work** pane.

Step 8 In the **Work** pane, click the **Operational** tab and click the **Traffic** subtab to view the atomic counter statistics.

Using SNMP

About SNMP

The Cisco Application Centric Infrastructure (ACI) provides extensive SNMPv1, v2, and v3 support, including Management Information Bases (MIBs) and notifications (traps). The SNMP standard allows any third-party applications that support the different MIBs to manage and monitor the ACI fabric.

SNMPv3 provides extended security. Each SNMPv3 device can be selectively enabled or disabled for SNMP service. In addition, each device can be configured with a method of handling SNMPv1 and v2 requests.

For more information about using SNMP, see the *Cisco ACI MIB Quick Reference*.

SNMP Access Support in ACI

SNMP support in ACI is as follows:

- SNMP read queries (Get, Next, Bulk, Walk) are supported by leaf and spine switches and by APIC.
- SNMP write commands (Set) are not supported by leaf and spine switches or by APIC.
- SNMP traps (v1, v2c, and v3) are supported by leaf and spine switches and by APIC.

Note ACI supports a maximum of 10 trap receivers.

- SNMPv3 is supported by leaf and spine switches and by APIC.

Table 1: SNMP Support Changes by Cisco APIC Release

Release	Description
1.2(2)	IPv6 support is added for SNMP trap destinations.
1.2(1)	SNMP support for the APIC controller is added. Previous releases support SNMP only for leaf and spine switches.

For the complete list of MIBs supported in ACI, see http://www.cisco.com/c/en/us/td/docs/switches/datacenter/aci/apic/sw/1-x/mib/list/mib-support.html.

Configuring SNMP

Configuring the SNMP Policy Using the GUI

This procedure configures and enables the SNMP policy on ACI switches.

Before You Begin

To allow SNMP communications, you must configure the following:

- Configure an out-of-band contract allowing SNMP traffic. SNMP traffic typically uses UDP port 161 for SNMP requests.
- Configure the APIC out-of-band IP addresses in the 'mgmt' tenant. Although the out-of-band addresses are configured during APIC setup, the addresses must be explicitly configured in the 'mgmt' tenant before the out-of-band contract will take effect.

Procedure

Step 1 In the menu bar, click **Fabric**.
Step 2 In the submenu bar, click **Fabric Policies**.
Step 3 In the **Navigation** pane, expand **Pod Policies**.
Step 4 Under **Pod Policies**, expand **Policies**.
Step 5 Right-click **SNMP** and choose **Create SNMP Policy**.

As an alternative to creating a new SNMP policy, you can edit the **default** policy fields in the same manner as described in the following steps.

Step 6 In the SNMP policy dialog box, perform the following actions:

a) In the **Name** field, enter an SNMP policy name.
b) In the **Admin State** field, select **Enabled**.
c) In the **Community Policies** table, click the + icon, enter a **Name** and click **Update**.
d) (Optional) In the **SNMP v3 Users** table, click the + icon, enter a **Name**, enter the user's authentication data, and click **Update**.
This step is needed only if SNMPv3 access is required.

Step 7 To configure allowed SNMP management stations, perform the following actions in the SNMP policy dialog box:

a) In the **Client Group Policies** table, click the + icon to open the **Create SNMP Client Group Profile** dialog box.
b) In the **Name** field, enter an SNMP client group profile name.
c) From the **Associated Management EPG** drop-down list, choose the management EPG.
d) In the **Client Entries** table, click the + icon.
e) Enter a client's name in the **Name** field, enter the client's IP address in the **Address** field, and click **Update**.

Step 8 Click **OK**.

Step 9 Click **Submit**.

Step 10 Under **Pod Policies**, expand **Policy Groups** and choose a policy group or right-click **Policy Groups** and choose **Create POD Policy Group**.
You can create a new pod policy group or you can use an existing group. The pod policy group can contain other pod policies in addition to the SNMP policy.

Step 11 In the pod policy group dialog box, perform the following actions:

a) In the **Name** field, enter a pod policy group name.
b) From the **SNMP Policy** drop-down list, choose the SNMP policy that you configured and click **Submit**.

Step 12 Under **Pod Policies**, expand **Profiles** and click **default**.

Step 13 In the **Work pane**, from the **Fabric Policy Group** drop-down list, choose the pod policy group that you created.

Step 14 Click **Submit**.

Step 15 Click **OK**.

Configuring an SNMP Trap Destination Using the GUI

This procedure configures the host information for an SNMP manager that will receive SNMP trap notifications.

Note ACI supports a maximum of 10 trap receivers. If you configure more than 10, some will not receive notifications.

Procedure

Step 1 In the menu bar, click **Admin**.

Step 2 In the submenu bar, click **External Data Collectors**.

Step 3 In the **Navigation** pane, expand **Monitoring Destinations**.

Step 4 Right-click **SNMP** and choose **Create SNMP Trap Destination Group**.

Step 5 In the **Create SNMP Trap Destination Group** dialog box, perform the following actions:

a) In the **Name** field, enter an SNMP destination name and click **Next**.

b) In the **Create Destinations** table, click the + icon to open the **Create SNMP Trap Destination** dialog box.

c) In the **Host Name/IP** field, enter an IP address or a fully qualified domain name for the destination host.

Note Cisco APIC Release 1.2(2) and later releases support IPv6 SNMP trap destinations.

d) Choose the **Port** number and SNMP **Version** for the destination.

e) For SNMP v1 or v2c destinations, enter one of the configured community names as **Security Name** and choose **noauth** as **v3 Security Level**.

f) For SNMP v3 destinations, enter one of the configured SNMP v3 user names as **Security Name** and choose the desired **v3 Security Level**.

g) From the **Management EPG** drop-down list, choose the management EPG.

h) Click **OK**.

i) Click **Finish**.

Configuring an SNMP Trap Source Using the GUI

This procedure selects and enables a source object within the fabric to generate SNMP trap notifications.

Procedure

Step 1 In the menu bar, click **Fabric**.

Step 2 In the submenu bar, click **Fabric Policies**.

Step 3 In the **Navigation** pane, expand **Monitoring Policies**.
You can create an SNMP source in the **Common Policy**, the **default** policy, or you can create a new monitoring policy.

Step 4 Expand the desired monitoring policy and choose **Callhome/SNMP/Syslog**.
If you chose the **Common Policy**, right-click **Common Policy**, choose **Create SNMP Source**, and follow the instructions below for that dialog box.

Step 5 In the **Work** pane, from the **Monitoring Object** drop-down list, choose **ALL**.

Step 6 From the **Source Type** drop-down list, choose **SNMP**.

Step 7 In the table, click the + icon to open the **Create SNMP Source** dialog box.

Step 8 In the **Create SNMP Source** dialog box, perform the following actions:

a) In the **Name** field, enter an SNMP policy name.

b) In the **Include** field, check all checkboxes for the desired type of notification (events, audit logs, faults).

c) From the **Min Severity** drop-down list, choose the **Info** severity level for triggering notifications.
d) From the **Dest Group** drop-down list, choose an existing destination for sending notifications or choose **Create SNMP Trap Destination Group** to create a new destination.
The steps for creating an SNMP destination group are described in a separate procedure.
e) Click **Submit**.

Monitoring the System Using SNMP

You can remotely monitor individual hosts (APIC or another host) and find out the state of any particular node.

You can check the system's CPU and memory usage using SNMP to find out if the CPU is spiking or not. The SNMP, a network management system, uses an SNMP client and accesses information over the APIC and retrieves information back from it.

You can remotely access the system to figure out if the information is in the context of the network management system and you can learn whether or not it is taking too much CPU or memory, or if there are any system or performance issues. Once you learn the source of the issue, you can check the system health and verify whether or not it is using too much memory or CPU).

Refer to the *Cisco ACI MIB Quick Reference Manual* for additional information.

Using SPAN

About SPAN

You can use the Switched Port Analyzer (SPAN) utility to perform detailed troubleshooting or to take a sample of traffic from a particular application host for proactive monitoring and analysis.

SPAN copies traffic from one or more ports, VLANs, or endpoint groups (EPGs) and sends the copied traffic to one or more destinations for analysis by a network analyzer. The process is nondisruptive to any connected devices and is facilitated in the hardware, which prevents any unnecessary CPU load.

You can configure SPAN sessions to monitor traffic received by the source (ingress traffic), traffic transmitted from the source (egress traffic), or both. By default, SPAN monitors all traffic, but you can configure filters to monitor only selected traffic.

Multinode SPAN

APIC traffic monitoring policies can SPAN policies at the appropriate places to track members of each application group and where they are connected. If any member moves, APIC automatically pushes the policy to the new leaf switch. For example, when an endpoint VMotions to a new leaf switch, the SPAN configuration automatically adjusts.

SPAN Guidelines and Restrictions

- Use SPAN for troubleshooting. SPAN traffic competes with user traffic for switch resources. To minimize the load, configure SPAN to copy only the specific traffic that you want to analyze.
- You cannot specify an l3extLIfP layer 3 subinterface as a SPAN source. You must use the entire port for monitoring traffic from external sources.
- Tenant and access SPAN use the encapsulated remote extension of SPAN (ERSPAN) type I, while fabric SPAN uses ERSPAN type II. For information regarding ERSPAN headers, refer to the IETF Internet Draft at this URL: https://tools.ietf.org/html/draft-foschiano-erspan-00.
- See the *Verified Scalability Guide for Cisco ACI* document for SPAN-related limits, such as the maximum number of active SPAN sessions.

Configuring a SPAN Session

This procedure shows how to configure a SPAN policy to forward replicated source packets to a remote traffic analyzer.

Procedure

Step 1 In the menu bar, click **Tenants**.

Step 2 In the submenu bar, click the tenant that contains the source endpoint.

Step 3 In the **Navigation** pane, expand the tenant, expand **Troubleshooting Policies**, and expand **SPAN**.

Step 4 Under **SPAN**, right-click **SPAN Destination Groups** and choose **Create SPAN Destination Group**.

Step 5 In the **Create SPAN Destination Group** dialog box, perform the following actions:

a) In the **Name** field, enter a name for the SPAN destination group.
b) In the **Create Destinations** table, click the + icon to open the **Create SPAN Destination** dialog box.
c) In the **Name** field, enter a name for the SPAN destination.
d) From the **Destination EPG** drop-down lists, choose or enter the destination tenant, application profile, or EPG to which replicated packets will be forwarded.
e) In the **Destination IP** field, enter the IP address of the remote server that will receive the replicated packets.
f) In the **Source IP Prefix** field, enter the base IP address of the IP subnet of the source packets.
g) (Optional) In the **Flow ID** field, increment or decrement the flow ID value of the SPAN packet.
h) (Optional) In the **TTL** field, increment or decrement the IP time-to-live (TTL) value of the packets in the SPAN traffic.
i) (Optional) In the **MTU** field, increment or decrement the MTU truncation size for the packets.
j) (Optional) In the **DSCP** field, increment or decrement the IP DSCP value of the packets in the SPAN traffic.
k) Click **OK** to save the SPAN destination.
l) Click **Submit** to save the SPAN destination group.

Step 6 Under **SPAN**, right-click **SPAN Source Groups** and choose **Create SPAN Source Group**.

Step 7 In the **Create SPAN Source Group** dialog box, perform the following actions:

a) In the **Name** field, enter a name for the SPAN source group.

b) From the **Destination Group** drop-down list, choose the SPAN destination group that you configured previously.
c) In the **Create Sources** table, click the + icon to open the **Create ERSPAN Source** dialog box.
d) In the **Name** field, enter a name for the source.
e) In the **Direction** field, choose the radio button based on whether you want to replicate and forward packets that are incoming to the source, outgoing from the source, or both incoming and outgoing.
f) From the **Source EPG** drop-down list, choose the EPG (identified by Tenant/ApplicationProfile/EPG) whose packets will be replicated and forwarded to the SPAN destination.
g) Click **OK** to save the SPAN source.
h) Click **Submit** to save the SPAN source group.

What to Do Next

Using a traffic analyzer at the SPAN destination, you can observe the data packets from the SPAN source EPG to verify the packet format, addresses, protocols, and other information.

Using Traceroute

About Traceroute

The traceroute tool is used to discover the routes that packets actually take when traveling to their destination. Traceroute identifies the path taken on a hop-by-hop basis and includes a time stamp at each hop in both directions. You can use traceroute to test the connectivity of ports along the path between the generating device and the device closest to the destination. If the destination cannot be reached, the path discovery traces the path up to the point of failure.

A traceroute that is initiated from the tenant endpoints shows the default gateway as an intermediate hop that appears at the ingress leaf switch.

Traceroute supports a variety of modes, including endpoint-to-endpoint, and leaf-to-leaf (tunnel endpoint, or TEP to TEP). Traceroute discovers all paths across the fabric, discovers point of exits for external endpoints, and helps to detect if any path is blocked.

Traceroute Guidelines and Restrictions

- When the traceroute source or destination is an endpoint, the endpoint must be dynamic and not static. Unlike a dynamic endpoint (fv:CEp), a static endpoint (fv:StCEp) does not have a child object (fv:RsCEpToPathEp) that is required for traceroute.
- See the *Verified Scalability Guide for Cisco ACI* document for traceroute-related limits.

Performing a Traceroute Between Endpoints

Procedure

Step 1 In the menu bar, click **Tenants**.

Step 2 In the submenu bar, click the tenant that contains the source endpoint.

Step 3 In the **Navigation** pane, expand the tenant and expand **Troubleshoot Policies**.

Step 4 Under **Troubleshoot Policies**, right-click **Endpoint-to-Endpoint Traceroute Policies** and choose **Create Endpoint-to-Endpoint Traceroute Policy**.

Step 5 In the **Create Endpoint-to-Endpoint Traceroute Policy** dialog box, perform the following actions:

a) In the **Name** field, enter a name for the traceroute policy.
b) In the **Source End Points** table, click the + icon to edit the traceroute source.
c) From the **Source MAC** drop-down list, choose or enter the MAC address of the source endpoint and click **Update**.
d) In the **Destination End Points** table, click the + icon to edit the traceroute destination.
e) From the **Destination MAC** drop-down list, choose or enter the MAC address of the destination endpoint and click **Update**.
f) In the **State** field, click the **Start** radio button.
g) Click **Submit** to launch the traceroute.

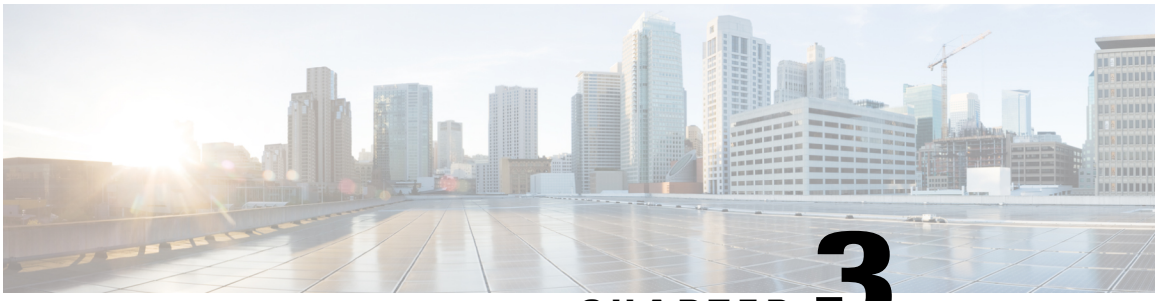
Step 6 In the **Navigation** pane or the **Traceroute Policies** table, click the traceroute policy.
The traceroute policy is displayed in the **Work** pane.

Step 7 In the **Work** pane, click the **Operational** tab, click the **Source End Points** tab, and click the **Results** tab.

Step 8 In the **Traceroute Results** table, verify the path or paths that were used in the trace.

Note More than one path might have been traversed from the source node to the destination node.

Note For readability, you can increase the width of one or more columns, such as the Name column.



CHAPTER 3

Provisioning Core ACI Fabric Services

This chapter contains the following sections:

Time Synchronization and NTP

Within the Cisco Application Centric Infrastructure (ACI) fabric, time synchronization is a crucial capability upon which many of the monitoring, operational, and troubleshooting tasks depend. Clock synchronization is important for proper analysis of traffic flows as well as for correlating debug and fault time stamps across multiple fabric nodes.

An offset present on one or more devices can hamper the ability to properly diagnose and resolve many common operational issues. In addition, clock synchronization allows for the full utilization of the atomic counter capability that is built into the ACI upon which the application health scores depend. Nonexistent or improper configuration of time synchronization does not necessarily trigger a fault or a low health score. You should configure time synchronization before deploying a full fabric or applications so as to enable proper usage of these features. The most widely adapted method for synchronizing a device clock is to use Network Time Protocol (NTP).

Prior to configuring NTP, consider what management IP address scheme is in place within the ACI fabric. There are two options for configuring management of all ACI nodes and Application Policy Infrastructure Controllers (APICs), in-band management and/or out-of-band management. Depending upon which management option is chosen for the fabric, configuration of NTP will vary. Another consideration in deploying time synchronization is where the time source is located. The reliability of the source must be carefully considered when determining if you will use a private internal clock or an external public clock.

In-Band and Out-of-Band Management NTP

Note

- Make sure the Management EPG is configured for the NTP servers, otherwise the servers will not get configured on the switches.
- See the Adding Management Access section in this guide for information about in-band management access and out-of-band management access.

- Out-of-band management NTP—When an ACI fabric is deployed with out-of-band management, each node of the fabric, inclusive of spines, leaves, and all members of the APIC cluster, is managed from outside the ACI fabric. This IP reachability will be leveraged so that each node can individually query the same NTP server as a consistent clock source. To configure NTP, a Date and Time policy must be created that references an out-of-band management endpoint group. Date and Time policies are confined to a single pod and must be deployed across all pods provisioned in the ACI fabric. Currently only one pod per ACI fabric is allowed.
- In-Band Management NTP—When an ACI fabric is deployed with in-band management, consider the reachability of the NTP server from within the ACI in-band management network. In-band IP addressing used within the ACI fabric is not reachable from anywhere outside the fabric. To leverage an NTP server external to the fabric with in-band management, construct a policy to enable this communication. The steps used to configure in-band management policies are identical to those used to establish an out-of-band management policy. The distinction is around how to allow the fabric to connect to the NTP server.

Configuring NTP Using the Advanced GUI

Procedure

Step 1 On the menu bar, choose **FABRIC** > **Fabric Policies**.

Step 2 In the **Navigation** pane, choose **Pod Policies** > **Policies**.

Step 3 In the **Work** pane, choose **Actions** > **Create Date and Time Policy**.

Step 4 In the **Create Date and Time Policy** dialog box, perform the following actions:

a) Enter a name for the policy to distinguish between the different NTP configurations in your environment. Click **Next**.

b) Click the + sign to specify the NTP server information (provider) to be used.

c) In the **Create Providers** dialog box, enter all relevant information, including the following fields: **Name**, **Description**, **Minimum Polling Intervals**, and **Maximum Polling Intervals**.

- If you are creating multiple providers, check the **Preferred** check box for the most reliable NTP source.
- In the Management EPG drop-down list, if the NTP server is reachable by all nodes on the fabric through out-of-band management, choose Out-of-Band. If you have deployed in-band management, see the details about In-Band Management NTP. Click **OK**.

Repeat the steps for each provider that you want to create.

Step 5 In the **Navigation** pane, choose **Pod Policies** > **Policy Groups**.

Step 6 In the **Work** pane, choose **Actions** > **Create Pod Policy Group**.

Step 7 In the **Create Pod Policy Group** dialog box, perform the following actions:

a) Enter a name for the policy group.

b) In the **Date Time Policy** field, from the drop down list, choose the NTP policy that you created earlier. Click **Submit**.
The pod policy group is created. Alternatively, you can use the default pod policy group.

Step 8 In the **Navigation** pane, choose **Pod Policies** > **Profiles**.

Step 9 In the **Work** pane, double-click the desired pod selector name.

Step 10 In the Properties area, from the **Fabric Policy Group** drop down list, choose the pod policy group you created. Click **Submit**.

Configuring NTP Using the REST API

Procedure

Step 1 Configure NTP.

Example:

```
POST url: https://APIC-IP/api/node/mo/uni/fabric/time-test.xml

<imdata totalCount="1">
    <datetimePol adminSt="enabled" authSt="disabled" descr="" dn="uni/fabric/time-CiscoNTPPol"
 name="CiscoNTPPol" ownerKey="" ownerTag="">
        <datetimeNtpProv descr="" keyId="0" maxPoll="6" minPoll="4" name="10.10.10.11"
preferred="yes">
            <datetimeRsNtpProvToEpg tDn="uni/tn-mgmt/mgmtp-default/inb-default"/>
        </datetimeNtpProv>
    </datetimePol>
</imdata>
```

Step 2 Add the default Date Time Policy to the pod policy group.

Example:

```
POST url: https://APIC-IP/api/node/mo/uni/fabric/funcprof/podpgrp-calo1/rsTimePol.xml

POST payload: <imdata totalCount="1">
<fabricRsTimePol tnDatetimePolName="CiscoNTPPol">
</fabricRsTimePol>
</imdata>
```

Step 3 Add the pod policy group to the default pod profile.

Example:

```
POST url:
https://APIC-IP/api/node/mo/uni/fabric/podprof-default/pods-default-typ-ALL/rspodPGrp.xml

payload: <imdata totalCount="1">
```

```
<fabricRsPodPGrp tDn="uni/fabric/funcprof/podpgrp-calo1" status="created">
</fabricRsPodPGrp>
</imdata>
```

Verifying NTP Policy Deployed to Each Node Using the NX-OS Style CLI

Procedure

Step 1 Log onto an APIC controller in the fabric using the SSH protocol.

Step 2 Attach to a node and check the NTP peer status, shown as follows:
```
apic1# fabric  node_name show ntp peer-status
```

Step 3 Repeat step 2 for different nodes in the fabric.

Verifying NTP Operation Using the GUI

Procedure

Step 1 On the menu bar, choose **FABRIC** > **Fabric Policies**.

Step 2 In the **Navigation** pane, choose **Pod Policies** > **Policies** > **Date and Time** > ***ntp_policy*** > ***server_name***. The *ntp_policy* is the previously created policy. An IPv6 address is supported in the Host Name/IP address field. If you enter a hostname and it has an IPv6 address set, you must implement the priority of IPv6 address over IPv4 address.

Step 3 In the **Work** pane, verify the details of the server.

Configuring a DHCP Relay Policy

A DHCP relay policy may be used when the DHCP client and server are in different subnets. If the client is on an ESX hypervisor with a deployed vShield Domain profile, then the use of a DHCP relay policy configuration is mandatory.

When a vShield controller deploys a Virtual Extensible Local Area Network (VXLAN), the hypervisor hosts create a kernel (vmkN, virtual tunnel end-point [VTEP]) interface. These interfaces need an IP address in the infrastructure tenant that uses DHCP. Therefore, you must configure a DHCP relay policy so that the APIC can act as the DHCP server and provide these IP addresses.

When an ACI fabric acts as a DHCP relay, it inserts the DHCP Option 82 (the DHCP Relay Agent Information Option) in DHCP requests that it proxies on behalf of clients. If a response (DHCP offer) comes back from a DHCP server without Option 82, it is silently dropped by the fabric. Therefore, when the ACI fabric acts

as a DHCP relay, DHCP servers providing IP addresses to compute nodes attached to the ACI fabric must support Option 82.

Configuring a DHCP Server Policy for the APIC Infrastructure Using the Advanced GUI

- To watch an example video of this task, see Videos Webpage.
- The port and the encapsulation used by the application EPG must belong to a physical or VM Manager (VMM) domain. If no such association with a domain is established, the APIC continues to deploy the EPG but raises a fault.
- Cisco APIC supports DHCP relay for both IPv4 and IPv6 tenant subnets. DHCP server addresses can be IPv4 or IPv6. DHCPv6 relay will occur only if IPv6 is enabled on the fabric interface and one or more DHCPv6 relay servers are configured.

Before You Begin

Make sure that Layer 2 or Layer 3 management connectivity is configured.

Procedure

Step 1 On the menu bar, choose **TENANTS** > **infra**. In the **Navigation** pane, under **Tenant infra**, expand **Networking** > **Protocol Policies** > **DHCP** > **Relay Policies**.

Step 2 Right-click **Relay Policies** and click **Create DHCP Relay Policy**.

Step 3 In the **Create DHCP Relay Policy** dialog box, perform the following actions:

a) In the **Name** field, enter the DHCP relay profile name (DhcpRelayP).

b) Expand **Providers**. In the **Create DHCP Provider** dialog box, in the **EPG Type** field, click the appropriate radio button depending upon where the DHCP server is connected.

c) In the **Application EPG** area, in the **Tenant** field, from the drop-down list, choose the tenant. (infra)

d) In the **Application Profile** field, from the drop-down list, choose the application. (access)

e) In the **EPG** field, from the drop-down list, choose the EPG. (default)

f) In the **DHCP Server Address** field, enter the IP address for the infra DHCP server. Click **Update**.

Note The infra DHCP IP address is the infra IP address of APIC1. You must enter the default IP address of 10.0.0.1 if deploying for vShield controller configuration.

g) Click **Submit**.

The DHCP relay policy is created.

Step 4 In the **Navigation** pane, expand **Networking** > **Bridge Domains** > **default** > **DHCP Relay Labels**.

Step 5 Right-click **DHCP Relay Labels**, and click **Create DHCP Relay Label**.

Step 6 In the **Create DHCP Relay Label** dialog box, perform the following actions:

a) In the **Scope** field, click the tenant radio button.
This action displays, in the **Name** field drop-down list, the DHCP relay policy created earlier.

b) In the **Name** field, from the drop-down list, choose the name of the DHCP policy created (DhcpRelayP).

c) Click **Submit**.

The DHCP server is associated with the bridge domain.

Step 7 In the **Navigation** pane, expand **Networking** > **Bridge Domains** > **default** > **DHCP Relay Labels** to view the DHCP server created.

Configuring a DHCP Server Policy for the APIC Infrastructure Using the NX-OS Style CLI

- The port and the encapsulation used by the application EPG must belong to a physical or VM Manager (VMM) domain. If no such association with a domain is established, the APIC continues to deploy the EPG but raises a fault.
- Cisco APIC supports DHCP relay for both IPv4 and IPv6 tenant subnets. DHCP server addresses can be IPv4 or IPv6. DHCPv6 relay will occur only if IPv6 is enabled on the fabric interface and one or more DHCPv6 relay servers are configured.

Before You Begin

Ensure that Layer 2 or Layer 3 connectivity is configured to reach the DHCP server address.

Procedure

Configure DHCP server policy settings for the APIC infrastructure traffic.

Example:

```
apic1(config)# tenant infra
apic1(config-tenant)# template dhcp relay policy DhcpRelayP
apic1(config-tenant-template-dhcp-relay)# ip address 10.0.0.1 tenant infra application access epg
default
apic1(config-tenant-template-dhcp-relay)# exit
apic1(config-tenant)# interface bridge-domain default
apic1(config-tenant-interface)# dhcp relay policy tenant DhcpRelayP
apic1(config-tenant-interface)# exit
```

Configuring a DHCP Server Policy for the APIC Infrastructure Using the REST API

- This task is a prerequisite for users who want to create a vShield Domain Profile.
- The port and the encapsulation used by the application EPG must belong to a physical or VM Manager (VMM) domain. If no such association with a domain is established, the APIC continues to deploy the EPG but raises a fault.
- Cisco APIC supports DHCP relay for both IPv4 and IPv6 tenant subnets. DHCP server addresses can be IPv4 or IPv6. DHCPv6 relay will occur only if IPv6 is enabled on the fabric interface and one or more DHCPv6 relay servers are configured.

Before You Begin

Make sure that Layer 2 or Layer 3 management connectivity is configured.

Procedure

Configure the APIC as the DHCP server policy for the infrastructure tenant.

Note This relay policy will be pushed to all the leaf ports that are connected hypervisors using the attach entity profile configuration. For details about configuring with attach entity profile, see the examples related to creating VMM domain profiles.

Example:

```
<!-- api/policymgr/mo/.xml -->
<polUni>

POST URL:
https://APIC-IP/api/mo/uni.xml


  <fvTenant name="infra">

    <dhcpRelayP name="DhcpRelayP"  owner="tenant">
      <dhcpRsProv tDn="uni/tn-infra/ap-access/epg-default" addr="10.0.0.1" />
    </dhcpRelayP>

    <fvBD name="default">
      <dhcpLbl name="DhcpRelayP" owner="tenant"/>
    </fvBD>

  </fvTenant>
</polUni>
```

Configuring a DNS Service Policy

A DNS policy is required to connect to external servers, for example AAA, RADIUS, vCenter, and services by hostname. A DNS service policy is a shared policy, so any tenant and VRF that uses this service must be configured with the specific DNS profile label. To configure a DNS policy for the ACI fabric, you must complete the following tasks:

- Ensure that the management EPG is configured for the DNS policy, otherwise this policy will not take into effect on the switches.
- Create a DNS profile (default) that contains the information about DNS providers and DNS domains.
- Associate the DNS profile (default or another DNS profile) name to a DNS label under the required tenant.

It is possible to configure a per-tenant, per-VRF DNS profile configuration. Additional DNS profiles can be created and applied to specific VRFs of specific tenants using the appropriate DNS label. For example, if you create a DNS profile with a name of acme, you can add a DNS label of acme to the appropriate **Networking** > **VRF** policy configuration in the tenants configuration.

Configuring External Destinations with an In-Band DNS Service Policy

Configure the external destinations for the services as follows:

Source	In-Band Management	Out-of-Band Management	External Server Location
APIC	IP address or Fully Qualified domain name (FQDN)	IP address or FQDN	Anywhere
Leaf switches	IP address	IP address or FQDN **Note** The DNS policy must specify the out-of-band management EPG for reachability of the DNS server.	Anywhere
Spine switches	IP address	IP address or FQDN **Note** The DNS policy must specify the out-of-band management EPG for reachability of the DNS server.	Directly connected to a leaf switch

The following is a list of external servers:

- Call Home SMTP server
- Syslog server
- SNMP Trap destination
- Statistics Export destination
- Configuration Export destination
- Techsupport Export destination
- Core Export destination

The recommended guidelines are as follows:

- The external servers must be atatched to the leaf access ports.
- Use in-band connectivity for the leaf switches to avoid extra cabling for the management port.
- Use out-of-band management connectivity for the spine switches. Connect this out-of-band network for spine switches to one of the leaf ports with in-band management virtual routing and forwarding (VRF) so that the spine switches and the leaf switches can reach the same set of external servers.
- Use IP addresses for the external servers.

Configuring a DNS Service Policy to Connect with DNS Providers Using the Advanced GUI

Note To watch an example video of this task, see Videos Webpage.

Before You Begin

Make sure that Layer 2 or Layer 3 management connectivity is configured.

Procedure

Step 1 On the menu bar, choose **FABRIC** > **Fabric Policies**. In the **Navigation** pane, expand **Global Policies** > **DNS Profiles**, and click the default DNS profile.

Step 2 In the **Work** pane, in the **Management EPG** field, from the drop-down list, choose the appropriate management EPG (default (Out-of-Band)).

Step 3 Expand **DNS Providers**, and perform the following actions:

a) In the **Address** field, enter the provider address.
b) In the **Preferred** column, check the check box if you want to have this address as the preferred provider. You can have only one preferred provider.
c) Click **Update**.
d) (Optional) To add a secondary DNS provider, expand **DNS Providers**, and in the **Address** field, type the provider address. Click **Update**.

Step 4 Expand **DNS Domains**, and perform the following actions:

a) In the **Name** field, enter the domain name (cisco.com).
b) In the **Default** column, check the check box to make this domain the default domain. You can have only one domain name as the default.
c) Click **Update**.
d) (Optional) To add a secondary DNS domain, expand **DNS Domains**. In the **Address** field, enter the secondary domain name. Click **Update**.

Step 5 Click **Submit**.
The DNS server is configured.

Step 6 On the menu bar, click **TENANTS** > **mgmt**.

Step 7 In the **Navigation** pane, expand **Networking** > **VRF** > **oob**, and click **oob**.

Step 8 In the **Work** pane, under **Properties**, in the **DNS labels** field, enter the appropriate DNS label (default). Click **Submit**.
The DNS profile label is now configured on the tenant and VRF.

Configuring a DNS Service Policy to Connect with DNS Providers Using the NX-OS Style CLI

Procedure

Step 1 In the NX-OS CLI, get into configuration mode, shown as follows:

Example:
```
apic1# configure
apic1(config)#
```

Step 2 Configure a DNS server policy.

Example:

```
apic1(config)# dns
apic1(config-dns)# address 172.21.157.5 preferred
apic1(config-dns)# address 172.21.157.6
apic1(config-dns)# domain company.local default
apic1(config-dns)# use-vrf oob-default
```

Step 3 Configure a DNS profile label on any VRF where you want to use the DNS profile.

Example:

```
apic1(config)# tenant mgmt
apic1(config-tenant)# vrf context oob
apic1(config-tenant-vrf)# dns label default
```

Configuring a DNS Service Policy to Connect with DNS Providers Using the REST API

Before You Begin

Make sure that Layer 2 or Layer 3 management connectivity is configured.

Procedure

Step 1 Configure the DNS service policy.

Example:
```
POST URL :
https://apic-IP/api/node/mo/uni/fabric.xml

<dnsProfile name="default">

    <dnsProv addr="172.21.157.5" preferred="yes"/>
    <dnsProv addr="172.21.157.6"/>
```

```
    <dnsDomain name="cisco.com" isDefault="yes"/>

    <dnsRsProfileToEpg tDn="uni/tn-mgmt/mgmtp-default/oob-default"/>

</dnsProfile>
```

Step 2 Configure the DNS label under the out-of-band management tenant.

Example:

```
POST URL: https://apic-IP/api/node/mo/uni/tn-mgmt/ctx-oob.xml
<dnsLbl name="default" tag="yellow-green"/>
```

Verifying that the DNS Profile is Configured and Applied to the Fabric Controller Switches Using the NX-OS Style CLI

Procedure

Step 1 Verify the configuration for the default DNS profile.

Example:

```
apic1# show running-config dns

# Command: show running-config dns
# Time: Sat Oct  3 00:23:52 2015
  dns
    address 172.21.157.5 preferred
    address 172.21.157.6
    domain company.local default
    use-vrf oob-default
    exit
```

Step 2 Verify the configurations for the DNS labels.

Example:

```
apic1# show running-config tenant mgmt vrf context oob

# Command: show running-config tenant mgmt vrf context oob
# Time: Sat Oct  3 00:24:36 2015
  tenant mgmt
    vrf context oob
      dns label default
      exit
    exit
```

Step 3 Verify that the applied configuration is operating on the fabric controllers.

Example:

```
apic1# cat /etc/resolv.conf
# Generated by IFC

nameserver 172.21.157.5
nameserver 172.21.157.6
```

Configuring Custom Certificate Guidelines

- Wildcard certificates (such as *.cisco.com, which is used across multiple devices) and its associated private key generated elsewhere are not supported on the APIC as there is no support to input the private key or password in the APIC.
- You must download and install the public intermediate and root CA certificates before generating a Certificate Signing Request (CSR). Although a root CA Certificate is not technically required to generate a CSR, Cisco requires the root CA certificate before generating the CSR to prevent mismatches between the intended CA authority and the actual one used to sign the CSR. The APIC verifies that the certificate submitted is signed by the configured CA.
- To use the same public and private keys for a renewed certificate generation, you must satisfy the following guidelines:
  - You must preserve the originating CSR as it contains the public key that pairs with the private key in the key ring.
  - The same CSR used for the originating certificate must be resubmitted for the renewed certificate if you want to re-use the public and private keys on the APIC.
  - Do not delete the original key ring when using the same public and private keys for the renewed certificate. Deleting the key ring will automatically delete the associated private key used with CSRs.

Configuring a Custom Certificate for Cisco ACI HTTPS Access Using the GUI

CAUTION: PERFORM THIS TASK ONLY DURING A MAINTENANCE WINDOW AS THERE IS A POTENTIAL FOR DOWNTIME. Expect a restart of all web servers in the fabric during this operation.

Before You Begin

Determine from which authority you will obtain the trusted certification so that you can create the appropriate Certificate Authority.

Procedure

Step 1 On the menu bar, choose **ADMIN** > **AAA**.

Step 2 In the **Navigation** pane, configure the certificate authority by performing the following actions:

a) Expand **Public Key Management**.
b) Right-click **Certificate Authorities**, and click **Create Certificate Authority**.
c) In the **Create Certificate Authority** dialog box, in the **Name** field, enter a name for the certificate authority.
d) In the **Certificate Chain** field, copy the intermediate and root certificates for the certificate authority that will sign the Certificate Signing Request (CSR) for the Cisco APIC.

The certificate should be in Base64 encoded X.509 (CER) format. The intermediate certificate is placed before the root CA certificate. It should look similar to the following example:

```
-----BEGIN CERTIFICATE-----
<Intermediate Certificate>
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
<Root CA Certificate>
-----END CERTIFICATE-----
```

e) Click **Submit**.

Step 3 In the **Navigation** pane, expand **Public Key Management** > **Key Rings**, and create a key ring by performing the following actions:

a) Right-click **Key Rings**, and click **Create Key Ring**.

a) In the **Create Key Ring** dialog box, in the **Name** field, enter a name.

b) In the **Certificate** field, do not add any content.

c) In the **Modulus** field, click the radio button for the desired key strength.

d) In the **Certificate Authority** field, from the drop-down list, choose the certificate authority that you created earlier. Click **Submit**.

In the **Work** pane, in the **Key Rings** area, the **Admin State** for the key ring created displays **Started**.

Note Do not delete the key ring. Deleting the key ring will automatically delete the associated private key used with CSRs.

Step 4 In the **Navigation** pane, right-click the key ring you created, and perform the following actions to generate a CSR.

a) Click **Create Certificate Request**.

b) In the **Subject** field, enter the fully qualified domain name (FQDN) of the Cisco APIC controller.

Note The /etc/hosts file must have an entry with the APIC controller IP address and its DNS name. The DNS name must match the subject in the certificate. Each APIC controller must have an entry in this file.

c) Enter the remaining fields as appropriate. Repeat this step (CSR) for each APIC controller and its appropriate certificate.

Note Check the online help information available in the **Create Certificate Request** dialog box for a description of the available parameters.

d) Click **Submit**.

The object is created and displayed in the **Navigation** pane under the key ring you created earlier. In the **Navigation** pane, click the object and in the **Work** pane, in the **Properties** area, in the **Request** field the CSR is displayed. Copy the contents from the field to submit to the **Certificate Authority**.

Step 5 In the **Navigation** pane, click the key ring you created and perform the following actions to install the signed certificate:

a) In the **Work** pane, in the **Certificate** field, paste the signed certificate received from the certificate authority.

b) Click **Submit**.

Note If the CSR was not signed by the Certificate Authority indicated in the key ring, or if the certificate has MS-DOS line endings, an error message is displayed and the certificate is not accepted. Remove the MS-DOS line endings.

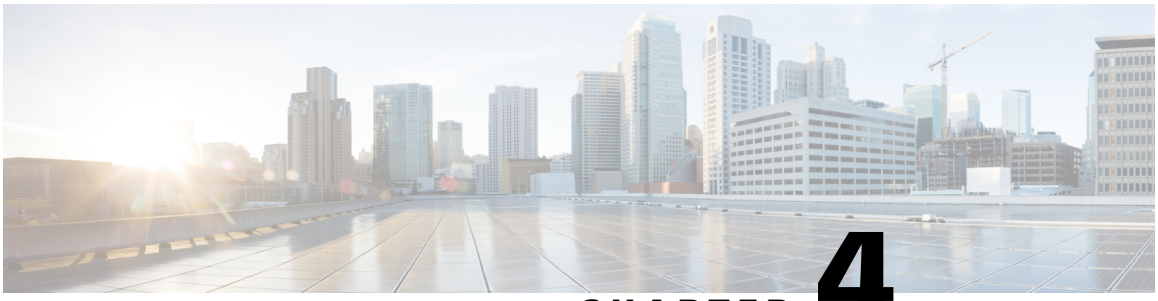
The key is verified, and in the **Work** pane, the **Admin State** changes to **Completed** and is now ready for use in the http policy.

Step 6 On the menu bar, choose **FABRIC** > **Fabric Policies**. In the Navigation pane, expand **Pod Policies** > **Policies** > **Communication** > **default**.

Step 7 In the **Work** pane, in the **Admin Key Ring** field, using the drop-down menu, choose the desired key ring. Click **Submit**.
All web servers restart. The certificate is activated, and the non-default key ring is associated with HTTPS access.

What to Do Next

You must remain aware of the expiry date of the certificate, and take action before it expires. To preserve the same key pair for the renewed certificate, you must preserve the CSR as it contains the public key that pairs with the private key in the key ring. Before the certificate expires, the same CSR must be resubmitted. Do not delete or create a new key ring as deleting the key ring will delete the private key stored internally on the APIC.



CHAPTER 4

ACI Fabric Access Layer 2 Connectivity

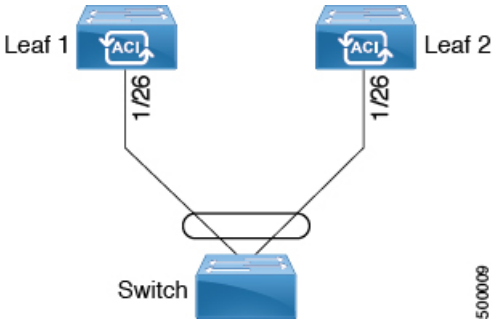
This chapter contains the following sections:

Layer 2 Workflows

ACI Virtual Port Channel Workflow

This workflow provides an overview of the steps required to configure a virtual port channel (VPC).

Figure 10: Virtual port channel configuration



1. Prerequisites

- Ensure that you have read/write access privileges to the infra security domain.
- Ensure that the target leaf switches with the necessary interfaces are available.

2. Configure the Virtual Port Channel

1 On the APIC menu bar, navigate to `Fabric > Access Policies > Quick Start`, and click **Configure an interface, PC, and VPC** to open the quick start wizard.

2 Provide the specifications for: the policy name, switch IDs and interfaces the virtual port channel will use; the Interface Policy group port speed, storm control, CDP, LLDP etc.; the Attached Device Type as an **External Bridged Device**, and specify the VLAN and domain that will be used.

3 Use the CLI **show int** command on the ACI leaf switches where the external switch is attached to verify that the switches and virtual port channel are configured accordingly.

Note: While this configuration enables hardware connectivity, no data traffic can flow without a valid application profile, EPG, and contract that is associated with this hardware configuration.

Configure the Application Profile

1. On the APIC menu bar, navigate to Tenant > <tenant name> > Quick Start, and click Create an application profile under the tenant quick start wizard.
2. Configure the endpoint groups (EPGs), contracts, bridge domain, subnet, and context.
3. Associate the application profile EPGs with the virtual port channel switch profile created above.

Suggested topics

For additional information, see the following topics:

- ACI Leaf Switch Virtual Port Channel Configuration Using the Advanced GUI, on page 103
- Configuring Virtual Port Channels in Leaf Nodes Using the NX-OS CLI, on page 106
- Creating an Application Profile Using the GUI, on page 141

Networking Domains

A fabric administrator creates domain policies that configure ports, protocols, VLAN pools, and encapsulation. These policies can be used exclusively by a single tenant, or shared. Once a fabric administrator configures domains in the ACI fabric, tenant administrators can associate tenant endpoint groups (EPGs) to domains.

These networking domain profiles can be configured:

- VMM domain profiles (`vmmDomP`) are required for virtual machine hypervisor integration.
- Physical domain profiles (`physDomP`) are typically used for bare metal server attachment and management access.
- Bridged outside network domain profiles (`l2extDomP`) are typically used to connect a bridged external network trunk switch to a leaf switch in the ACI fabric.
- Routed outside network domain profiles (`l3extDomP`) are used to connect a router to a leaf switch in the ACI fabric.

A domain is configured to be associated with a VLAN pool. EPGs are then configured to use the VLANs associated with a domain.

Note EPG port and VLAN configurations must match those specified in the domain infrastructure configuration with which the EPG associates. If not, the APIC will raise a fault. When such a fault occurs, verify that the domain infrastructure configuration matches the EPG port and VLAN configurations.

Attachable Entity Profile

The ACI fabric provides multiple attachment points that connect through leaf ports to various external entities such as bare metal servers, virtual machine hypervisors, Layer 2 switches (for example, the Cisco UCS fabric interconnect), or Layer 3 routers (for example Cisco Nexus 7000 Series switches). These attachment points can be physical ports, FEX ports, port channels, or a virtual port channel (vPC) on leaf switches.

An Attachable Entity Profile (AEP) represents a group of external entities with similar infrastructure policy requirements. The infrastructure policies consist of physical interface policies that configure various protocol options, such as Cisco Discovery Protocol (CDP), Link Layer Discovery Protocol (LLDP), Maximum Transmission Unit (MTU), or Link Aggregation Control Protocol (LACP).

An AEP is required to deploy VLAN pools on leaf switches. Encapsulation blocks (and associated VLANs) are reusable across leaf switches. An AEP implicitly provides the scope of the VLAN pool to the physical infrastructure.

The following AEP requirements and dependencies must be accounted for in various configuration scenarios, including network connectivity and VMM domains:

- The AEP defines the range of allowed VLANS but it does not provision them. No traffic flows unless an EPG is deployed on the port. Without defining a VLAN pool in an AEP, a VLAN is not enabled on the leaf port even if an EPG is provisioned.
- A particular VLAN is provisioned or enabled on the leaf port that is based on EPG events either statically binding on a leaf port or based on VM events from external controllers such as VMware vCenter or Microsoft Azure Service Center Virtual Machine Manager (SCVMM).

A virtual machine manager (VMM) domain automatically derives physical interface policies from the interface policy groups of an AEP.

An override policy at the AEP can be used to specify a different physical interface policy for a VMM domain. This policy is useful in scenarios where a VM controller is connected to the leaf switch through an intermediate Layer 2 node, and a different policy is desired at the leaf switch and VM controller physical ports. For example, you can configure LACP between a leaf switch and a Layer 2 node. At the same time, you can disable LACP between the VM controller and the Layer 2 switch by disabling LACP under the AEP override policy.

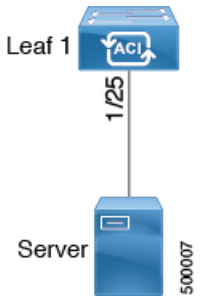
Configuration of Leaf Switch Physical Ports

Configuring Leaf Switch Physical Ports Using the Advanced GUI

The procedure below uses a Quick Start wizard.

Note This procedure provides the steps for attaching a server to an ACI leaf switch interface. The steps would be the same for attaching other kinds of devices to an ACI leaf switch interface.

Figure 11: Switch Interface Configuration for Bare Metal Server



Before You Begin

- The ACI fabric is installed, APIC controllers are online, and the APIC cluster is formed and healthy.
- An APIC fabric administrator account is available that will enable creating the necessary fabric infrastructure configurations.
- The target leaf switches are registered in the ACI fabric and available.

Procedure

Step 1 On the APIC menu bar, navigate to **Fabric** > **Access Policies** > **Quick Start**, and click *Configure an interface, PC, and VPC*.

Step 2 In the **Select Switches To Configure Interfaces** work area, click the large + to select switches to configure. In the *Switches* section, click the + to add switch ID(s) from the drop-down list of available switch IDs and click **Update**.

Step 3 Click the large + to configure switch interfaces.
The interface policy group is a named policy that specifies the group of interface policies you will apply to the selected interfaces of the switch. Examples of interface policies include Link Level Policy (for example, 1gbit port speed), Storm Control Interface Policy, and so forth.

Note The *Attached Dervice Type* domain is required for enabling an EPG to use the interfaces specified in the switch profile.

a) Specify *individual* as the interface type to use.
b) Specify the interface ID to use.
c) Specify the interface policies to use.
d) Specify the attached device type to use. Choose Bare Metal for connecting bare metal servers. Bare metal uses the phys domain type.
e) Click **Save** to update the policy details, then click **Submit** to submit the switch profile to the APIC.
The APIC creates the switch profile, along with the interface, selector, and attached device type policies.

Verification: Use the CLI **show int** command on the switch where the server is attached to verify that the switch interface is configured accordingly.

What to Do Next

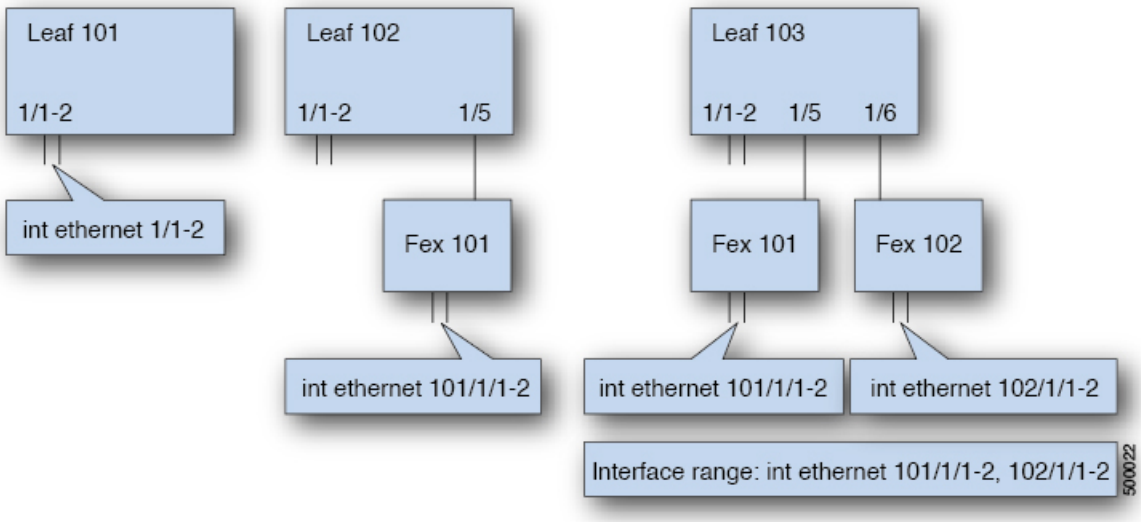
This completes the basic leaf interface configuration steps.

Note While this configuration enables hardware connectivity, no data traffic can flow without a valid application profile, EPG, and contract that is associated with this hardware configuration.

Configuring Physical Ports in Leaf Nodes Using the NX-OS CLI

The commands in the following examples create many managed objects (MOs) in the ACI policy model that are fully compatible with the REST API/SDK and GUI. However, the CLI user can focus on the intended network configuration instead of ACI model internals.

The following figure shows examples of Ethernet ports directly on leaf nodes or FEX modules attached to leaf nodes and how each is represented in the CLI. For FEX ports, the *fex-id* is included in the naming of the port itself as in **ethernet 101/1/1**. While describing an interface range, the **ethernet** keyword need not be repeated as in NX-OS. Example: **interface ethernet 101/1/1-2, 102/1/1-2**.



- Leaf node ID numbers are global.
- The *fex-id* numbers are local to each leaf.
- Note the space after the keyword **ethernet**.

Procedure

	Command or Action	Purpose
Step 1	**configure** **Example:** `apic1# configure`	Enters global configuration mode.
Step 2	**leaf** *node-id* **Example:** `apic1(config)# leaf 102`	Specifies the leaf or leafs to be configured. The *node-id* can be a single node ID or a range of IDs, in the form *node-id1-node-id2*, to which the configuration will be applied.
Step 3	**interface** *type* **Example:** `apic1(config-leaf)# interface ethernet 1/2`	Specifies the interface that you are configuring. You can specify the interface type and identity. For an Ethernet port, use "ethernet slot / port."
Step 4	**fex associate** *node-id* **Example:** `apic1(config-leaf-if)# fex associate 101`	(Optional) If the interface or interfaces to be configured are FEX interfaces, you must use this command to attach the FEX module to a leaf node before configuration. **Note** This step is required before creating a port-channel using FEX ports.
Step 5	**speed** *speed* **Example:** `apic1(config-leaf-if)# speed 10G`	The speed setting is shown as an example. At this point you can configure any of the interface settings shown in the table below.

The following table shows the interface settings that can be configured at this point.

Command	Purpose
[no] shut	Shut down physical interface
[no] speed *speedValue*	Set the speed for physical interface
[no] link debounce time *time*	Set link debounce
[no] negotiate auto	Configure negotiate
[no] cdp enable	Disable/enable Cisco Discovery Protocol (CDP)
[no] mcp enable	Disable/enable Mis-cabling Protocol (MCP)
[no] lldp transmit	Set the transmit for physical interface
[no] lldp receive	Set the LLDP receive for physical interface

Command	Purpose
spanning-tree {bpduguard | bpdufilter} {enable | disable}	Configure spanning tree BPDU
[no] storm-control level *percentage* [burst-rate *percentage*]	Storm-control configuration (percentage)
[no] storm-control pps *packets-per-second* burst-rate *packets-per-second*	Storm-control configuration (packets-per-second)

Examples

Configure one port in a leaf node. The following example shows how to configure the interface eth1/2 in leaf 101 for the following properties: speed, cdp, and admin state.

```
apic1# configure
apic1(config)# leaf 101
apic1(config-leaf)# interface ethernet 1/2
apic1(config-leaf-if)# speed 10G
apic1(config-leaf-if)# cdp enable
apic1(config-leaf-if)# no shut
```

Configure multiple ports in multiple leaf nodes. The following example shows the configuration of speed for interfaces eth1/1-10 for each of the leaf nodes 101-103.

```
apic1(config)# leaf 101-103
apic1(config-leaf)# interface eth 1/1-10
apic1(config-leaf-if)# speed 10G
```

Attach a FEX to a leaf node. The following example shows how to attach a FEX module to a leaf node. Unlike in NX-OS, the leaf port Eth1/5 is implicitly configured as fabric port and a FEX fabric port-channel is created internally with the FEX uplink port(s). In ACI, the FEX fabric port-channels use default configuration and no user configuration is allowed.

Note This step is required before creating a port-channel using FEX ports, as described in the next example.

```
apic1(config)# leaf 102
apic1(config-leaf)# interface eth 1/5
apic1(config-leaf-if)# fex associate 101
```

Configure FEX ports attached to leaf nodes. This example shows configuration of speed for interfaces eth1/1-10 in FEX module 101 attached to each of the leaf nodes 102-103. The FEX ID 101 is included in the port identifier. FEX IDs start with 101 and are local to a leaf.

```
apic1(config)# leaf 102-103
apic1(config-leaf)# interface eth 101/1/1-10
apic1(config-leaf-if)# speed 1G
```

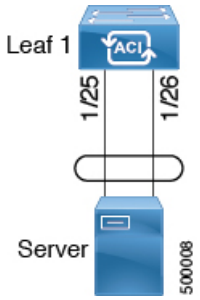
Configuration of Leaf Switch Port Channels

ACI Leaf Switch Port Channel Configuration Using the Advanced GUI

The procedure below uses a Quick Start wizard.

Note This procedure provides the steps for attaching a server to an ACI leaf switch interface. The steps would be the same for attaching other kinds of devices to an ACI leaf switch interface.

Figure 12: Switch Port Channel Configuration



Before You Begin

- The ACI fabric is installed, APIC controllers are online, and the APIC cluster is formed and healthy.
- An APIC fabric administrator account is available that will enable creating the necessary fabric infrastructure configurations.
- The target leaf switches are registered in the ACI fabric and available.

Procedure

Step 1 On the APIC menu bar, navigate to **Fabric** > **Access Policies** > **Quick Start**, and click *Configure an interface, PC, and VPC*.

Step 2 In the **Select Switches To Configure Interfaces** work area, click the large + to select switches to configure. In the *Switches* section, click the + to add switch ID(s) from the drop-down list of available switch IDs and click **Update**.

Step 3 Click the large + to configure switch interfaces.
The interface policy group is a named policy that specifies the group of interface policies you will apply to the selected interfaces of the switch. Examples of interface policies include Link Level Policy (for example, 1gbit port speed), Storm Control Interface Policy, and so forth.

Note The *Attached Dervice Type* is required for enabling an EPG to use the interfaces specified in the switch profile.

a) Specify *pc* as the interface type to use.
b) Specify the interface IDs to use.
c) Specify the interface policies to use.
d) Specify the attached device type to use. Choose Bare Metal for connecting bare metal servers. Bare metal uses the phys domain type.
e) Click **Save** to update the policy details, then click **Submit** to submit the switch profile to the APIC. The APIC creates the switch profile, along with the interface, selector, and attached device type policies.

Verification: Use the CLI **show int** command on the switch where the server is attached to verify that the switch interface is configured accordingly.

What to Do Next

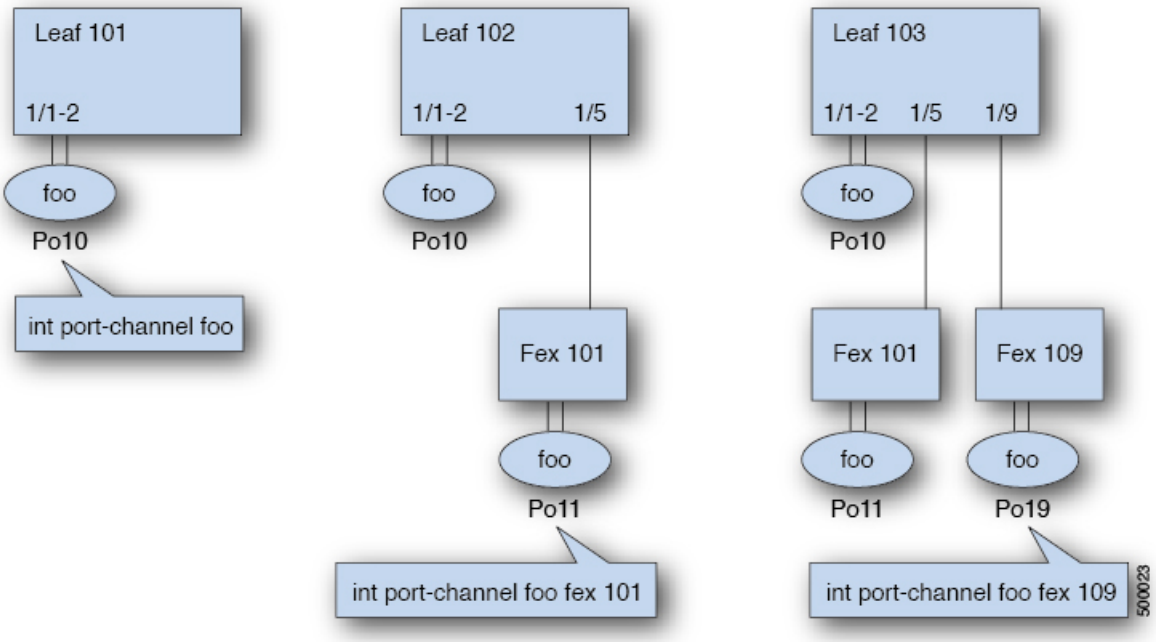
This completes the port channel configuration steps.

Note While this configuration enables hardware connectivity, no data traffic can flow without a valid application profile, EPG, and contract that is associated with this hardware configuration.

Configuring Port Channels in Leaf Nodes Using the NX-OS CLI

Port-channels are logical interfaces in NX-OS used to aggregate bandwidth for multiple physical ports and also for providing redundancy in case of link failures. In NX-OS, port-channel interfaces are identified by user-specified numbers in the range 1 to 4096 unique within a node. Port-channel interfaces are either configured explicitly (using interface port-channel command) or created implicitly (using channel-group command). The configuration of the port-channel interface is applied to all the member ports of the port-channel. There are certain compatibility parameters (speed, for example) that cannot be configured on the member ports.

In the ACI model, port-channels are configured as logical entities identified by a name to represent a collection of policies that can be assigned to set of ports in one or more leaf nodes. Such assignment creates one port-channel interface in each of the leaf nodes identified by an auto-generated number in the range 1 to 4096 within the leaf node, which may be same or different among the nodes for the same port-channel name. The membership of these port-channels may be same or different as well. When port-channel is created on the FEX ports, the same port-channel name can be used to create one port-channel interface in each of the FEX attached to the leaf node. Thus, it is possible to create up to N+1 unique port-channel interfaces (identified by the auto-generated port-channel numbers) for each leaf node attached to N FEX modules. This is illustrated with the examples below. Port-channels on the FEX ports are identified by specifying the *fex-id* along with the port-channel name (**interface port-channel foo fex 101**, for example).



- N+1 instances per leaf of port-channel foo are possible when each leaf is connected to N FEX nodes.
- Leaf ports and FEX ports cannot be part of the same port-channel instance.
- Each FEX node can have only one instance of port-channel foo.

Procedure

	Command or Action	Purpose
Step 1	**configure** **Example:** `apic1# configure`	Enters global configuration mode.
Step 2	**template port-channel** *channel-name* **Example:** `apic1(config)# template port-channel foo`	Creates a new port-channel or configures an existing port-channel (global configuration).
Step 3	**channel-mode active** **Example:** `apic1(config-if)# channel-mode active`	**Note** The **channel-mode** command is equivalent to the mode option in the channel-group command in NX-OS. In ACI, however, this is supported for the port-channel (not on a member port).
Step 4	**exit** **Example:** `apic1(config-if)# exit`	Returns to configure mode.

	Command or Action	Purpose
Step 5	**leaf** *node-id* **Example:** `apic1(config)# leaf 101`	Specifies the leaf or leafs to be configured. The *node-id* can be a single node ID or a range of IDs, in the form *node-id1-node-id2*, to which the configuration will be applied.
Step 6	**interface** *type* **Example:** `apic1(config-leaf)# interface ethernet 1/1-2`	Specifies the interface or range of interfaces that you are configuring to the port-channel.
Step 7	**[no] channel-group** *channel-name* **Example:** `apic1(config-leaf-if)# channel-group foo`	Assigns the interface or range of interfaces to the port-channel. Use the keyword **no** to remove the interface from the port-channel. To change the port-channel assignment on an interface, you can enter the **channel-group** command without first removing the interface from the previous port-channel.

Step 8	**lacp port-priority** *priority*
	
	Example:
	`apic1(config-leaf-if)# lacp port-priority 1000`
	`apic1(config-leaf-if)# lacp rate fast`
		(Optional)
		This setting and other per-port LACP properties can be applied to member ports of a port-channel at this point.
		Note In the ACI model, these commands are allowed only after the ports are member of a port channel. If a port is removed from a port channel, configuration of these per-port properties are removed as well.

The following table shows various commands for global configurations of port channel properties in the ACI model. These commands can also be used for configuring overrides for port channels in a specific leaf in the (config-leaf-if) CLI mode. The configuration made on the port-channel is applied to all member ports.

CLI Syntax	Feature
[no] speed <speedValue>	Set the speed for port-channel
[no] link debounce time <time>	Set Link Debounce for port-channel
[no] negotiate auto	Configure Negotiate for port-channel
[no] cdp enable	Disable/Enable cdp for port-channel
[no] mcp enable	Disable/Enable mcp for port-channel
[no] lldp transmit	Set the transmit for port-channel
[no] lldp receive	Set the lldp receive for port-channel

CLI Syntax	Feature
spanning-tree <bpduguard | bpdufilter> <enable | disable>	Configure spanning tree bpdu
[no] storm-control level <percentage> [burst-rate <percentage>]	Storm-control configuration (percentage)
[no] storm-control pps <packet-per-second> burst-rate <packets-per-second>	Storm-control configuration (packets-per-second)
[no] channel-mode { active | passive | on| mac-pinning }	LACP mode for the link in port-channel l
[no] lacp min-links <value>	Set minimum number of links
[no] lacp max-links <value>	Set maximum number of links
[no] lacp fast-select-hot-standby	LACP fast select for hot standby ports
[no] lacp graceful-convergence	LACP graceful convergence
[no] lacp load-defer	LACP load defer member ports
[no] lacp suspend-individual	LACP individual Port suspension
[no] lacp port-priority	LACP port priority
[no] lacp rate	LACP rate

Examples

Configure a port channel (global configuration). A logical entity foo is created that represents a collection of policies with two configurations: speed and channel mode. More properties can be configured as required.

Note The channel mode command is equivalent to the mode option in the channel group command in NX-OS. In ACI, however, this supported for the port-channel (not on member port).

```
(config)# template port-channel foo
(config-if)# speed 10G
(config-if)# channel-mode active
```

Configure ports to a port-channel in a FEX. In this example, port channel foo is assigned to ports Ethernet 1/1-2 in FEX 101 attached to leaf node 102 to create an instance of port channel foo. The leaf node will auto-generate a number, say 1002 to identify the port channel in the switch. This port channel number would be unique to the leaf node 102 regardless of how many instance of port channel foo are created.

Note The configuration to attach the FEX module to the leaf node must be done before creating port channels using FEX ports.

```
(config)# leaf 102
(config-leaf)# interface ethernet 101/1/1-2
(config-leaf-if)# channel-group foo
```

In Leaf 102, this port channel interface can be referred to as interface port-channel foo FEX 101.

```
(config)# leaf 102
(config-leaf)# interface port-channel foo fex 101
(config-leaf)# shut
```

Configure ports to a port channel in multiple leaf nodes. In this example, port channel foo is assigned to ports Ethernet 1/1-2 in each of the leaf nodes 101-103. The leaf nodes will auto generate a number unique in each node (which may be same or different among nodes) to represent the port-channel interfaces.

```
(config)# leaf 101-103
(config-leaf)# interface ethernet 1/1-2
(config-leaf-if)# channel-group foo
```

Add members to port channels. This example would add two members eth1/3-4 to the port-channel in each leaf node, so that port-channel foo in each node would have members eth 1/1-4.

```
(config)# leaf 101-103
(config-leaf)# interface ethernet 1/3-4
(config-leaf-if)# channel-group foo
```

Remove members from port channels. This example would remove two members eth1/2, eth1/4 from the port channel foo in each leaf node, so that port channel foo in each node would have members eth 1/1, eth1/3.

```
 (config)# leaf 101-103
(config-leaf)# interface eth 1/2,1/4
(config-leaf-if)# no channel-group foo
```

Configure port-channel with different members in multiple leaf nodes. This example shows how to use the same port-channel foo policies to create a port-channel interface in multiple leaf nodes with different member ports in each leaf. The port-channel numbers in the leaf nodes may be same or different for the same port-channel foo. In the CLI, however, the configuration will be referred as interface port-channel foo. If the port-channel is configured for the FEX ports, it would be referred to as interface port-channel foo fex <fex-id>.

```
(config)# leaf 101
(config-leaf)# interface ethernet 1/1-2
(config-leaf-if)# channel-group foo
(config-leaf-if)# exit
(config-leaf)# exit
(config)# leaf 102
(config-leaf)# interface ethernet 1/3-4
(config-leaf-if)# channel-group foo
(config-leaf-if)# exit
(config-leaf)# exit
(config)# leaf 103
(config-leaf)# interface ethernet 1/5-8
(config-leaf-if)# channel-group foo
(config-leaf-if)# exit
(config-leaf)# interface ethernet 101/1/1-2
(config-leaf-if)# channel-group foo
```

Configure per port properties for LACP. This example shows how to configure member ports of a port-channel for per-port properties for LACP.

Note In ACI model, these commands are allowed only after the ports are member of a port channel. If a port is removed from a port channel, configuration of these per-port properties would be removed as well.

```
(config)# leaf 101
(config-leaf)# interface ethernet 1/1-2
(config-leaf-if)# channel-group foo
(config-leaf-if)# lacp port-priority 1000
(config-leaf-if)# lacp rate fast
```

Configure admin state for port channels. In this example, a port-channel foo is configured in each of the leaf nodes 101-103 using the channel-group command. The admin state of port-channel(s) can be configured in each leaf using the port-channel interface. In ACI model, the admin state of the port-channel cannot be configured in the global scope.

```
// create port-channel foo in each leaf
(config)# leaf 101-103
(config-leaf)# interface ethernet 1/3-4
(config-leaf-if)# channel-group foo

// configure admin state in specific leaf
(config)# leaf 101
(config-leaf)# interface port-channel foo
(config-leaf-if)# shut
```

Override config is very helpful to assign specific vlan-domain, for example, to the port-channel interfaces in each leaf while sharing other properties.

```
// configure a port channel global config
(config)# interface port-channel foo
(config-if)# speed 1G
(config-if)# channel-mode active

// create port-channel foo in each leaf
(config)# leaf 101-103
(config-leaf)# interface ethernet 1/1-2
(config-leaf-if)# channel-group foo

// override port-channel foo in leaf 102
(config)# leaf 102
(config-leaf)# interface port-channel foo
(config-leaf-if)#  speed 10G
(config-leaf-if)#  channel-mode on
(config-leaf-if)#  vlan-domain dom-foo
```

This example shows how to change port channel assignment for ports using the channel-group command. There is no need to remove port channel membership before assigning to other port channel.

```
(config)# leaf 101-103
(config-leaf)# interface ethernet 1/3-4
(config-leaf-if)# channel-group foo
(config-leaf-if)# channel-group bar
```

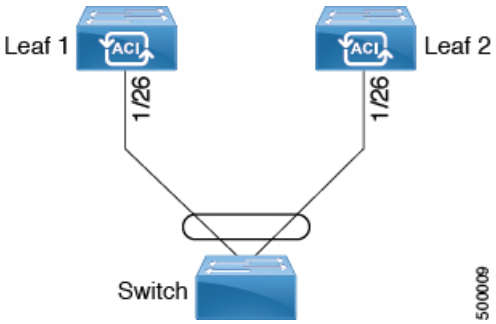
Configuration of Leaf Switch Virtual Port Channels

ACI Leaf Switch Virtual Port Channel Configuration Using the Advanced GUI

The procedure below uses a Quick Start wizard.

Note This procedure provides the steps for attaching a trunked switch to a ACI leaf switch virtual port channel. The steps would be the same for attaching other kinds of devices to an ACI leaf switch interface.

Figure 13: Switch Virtual Port Channel Configuration



Note LACP sets a port to the suspended state if it does not receive an LACP PDU from the peer. This can cause some servers to fail to boot up as they require LACP to logically bring-up the port. You can tune behavior to individual use by disabling **LACP suspend individual**. To do so, create a port channel policy in your vPC policy group, and after setting the mode to LACP active, remove **Suspend Individual Port**. Now the ports in the vPC will stay active and continue to send LACP packets.

Note Adaptive Load Balancing (ALB) (based on ARP Negotiation) across virtual port channels is not supported in the ACI.

Before You Begin

- The ACI fabric is installed, APIC controllers are online, and the APIC cluster is formed and healthy.
- An APIC fabric administrator account is available that will enable creating the necessary fabric infrastructure configurations.
- The target leaf switches are registered in the ACI fabric and available.

Note When creating a VPC domain between two leaf switches, both switches must be in the same switch generation, one of the following:

- Generation 1 - Cisco Nexus N9K switches without “EX” on the end of the switch name; for example, N9K-9312TX
- Generation 2 – Cisco Nexus N9K switches with “EX” on the end of the switch model name; for example, N9K-93108TC-EX

  Switches such as these two are not compatible VPC peers. Instead, use switches of the same generation.

Procedure

Step 1 On the APIC menu bar, navigate to **Fabric** > **Access Policies** > **Quick Start**, and click *Configure an interface, PC, and VPC*.

Step 2 In the *Configure an interface, PC, and VPC* work area, click the large + to select switches.
The **Select Switches To Configure Interfaces** work area opens.

Step 3 Select switch IDs from the drop-down list, name the profile, then click **Save**.
The saved policy displays in the *Configured Switch Interfaces* list.

Step 4 Configure the *Interface Policy Group* and *Attached Device Type* that the virtual port channel will use for the selected switches.
The interface policy group is a named policy that specifies the group of interface policies you will apply to the selected interfaces of the switch. Examples of interface policies include Link Level Policy (for example, 1gbit port speed), Storm Control Interface Policy, and so forth.

Note The *Attached Dervice Type* domain is required for enabling an EPG to use the interfaces specified in the switch profile.

a) Specify *vpc* the interface type (individual, pc, or vpc) to use.
b) Specify the interface IDs to use.
c) Specify the interface policies to use.
d) Specify the attached device type to use. Choose External Bridged Devices for connecting a switch.
e) Specify the *Domain*, and *VLAN Range*.
f) Click **Save** to update the policy details, then click **Submit** to submit the switch profile to the APIC.
The APIC creates the switch profile, along with the interface, selector, and attached device type policies.

Verification: Use the CLI **show int** command on the leaf switches where the external switch is attached to verify that the vpc is configured accordingly.

What to Do Next

This completes the switch virtual port channel configuration steps.

Note While this configuration enables hardware connectivity, no data traffic can flow without a valid application profile, EPG, and contract that is associated with this hardware configuration.

Configuring Virtual Port Channels in Leaf Nodes Using the NX-OS CLI

A virtual Port Channels (vPC) is an enhancement to port-channels that allows connection of a host or switch to two upstream leaf nodes to improve bandwidth utilization and availability. In NX-OS, vPC configuration is done in each of the two upstream switches and configuration is synchronized using peer link between the switches. The ACI model does not require a peer link and vPC configuration can be done globally for both the upstream leaf nodes. A global configuration mode called **vpc context** is introduced in ACI and vPC interfaces are represented using a type **interface vpc** that allows global configuration applicable to both leaf nodes.

Two different topologies are supported for vPC in the ACI model: vPC using leaf ports and vPC over FEX ports. It is possible to create many vPC interfaces between a pair of leaf nodes and similarly, many vPC interfaces can be created between a pair of FEX modules attached to the leaf node pairs in a straight-through topology.

vPC considerations include:

- The vPC name used is unique between leaf node pairs. For example, only one vPC 'foo' can be created per leaf pair (with or without FEX).
- Leaf ports and FEX ports cannot be part of the same vPC.
- Each FEX module can be part of only one instance of vPC foo.
- vPC context allows configuration
- The vPC context mode allows configuration of all vPCs for a given leaf pair. For vPC over FEX, the *fex-id* pairs must be specified either for the vPC context or along with the vPC interface, as shown in the following two alternative examples.

```
(config)# vpc context leaf 101 102
(config-vpc)# interface vpc bar fex 101 101
```

or

```
(config)# vpc context leaf 101 102 fex 101 101
(config-vpc)# interface vpc bar
```

In the ACI model, vPC configuration is done in the following steps (as shown in the examples below):

1. vPC domain configuration (global config)
2. Port-channel configuration for vPC (global config)
3. Configure ports to vPC in leaf nodes
4. Configure L2, L3 for vPC in the vpc context

Procedure

	Command or Action	Purpose
Step 1	**configure** **Example:** `apic1# configure`	Enters global configuration mode.
Step 2	**vpc domain explicit** *domain-id* **leaf** *node-id1 node-id2* **Example:** `apic1(config)# vpc domain explicit 1 leaf 101 102`	Configures a vPC domain between a pair of leaf nodes. You can specify the vPC domain ID in the explicit mode along with the leaf node pairs. Alternative commands to configure a vPC domain are as follows: • **vpc domain** [**consecutive** | **reciprocal**] The consecutive and reciprocal options allow auto configuration of a vPC domain across all leaf nodes in the ACI fabric. • **vpc domain consecutive** *domain-start* **leaf** *start-node end-node* This command configures a vPC domain consecutively for a selected set of leaf node pairs.
Step 3	**peer dead interval** *value* **Example:** `apic1(config-vpc)# peer dead interval 10`	The interval between hello packets from a neighbor before the router declares the neighbor as down. This value must be the same for all networking devices on a specific network. Specifying a smaller dead interval (seconds) will give faster detection of a neighbor being down and improve convergence, but might cause more routing instability.
Step 4	**exit** **Example:** `apic1(config-vpc)# exit`	Returns to global configuration mode.
Step 5	**template port-channel** *channel-name* **Example:** `apic1(config)# template port-channel foo`	Creates a new port-channel or configures an existing port-channel (global configuration). All vPC are configured as port-channels in each leaf pair. The same port-channel name must be used in a leaf pair for the same vPC. This port-channel can be used to create a vPC among one or more pairs of leaf nodes. Each leaf node will have only one instance of this vPC.
Step 6	**lacp mode active** **Example:** `apic1(config-if)# lacp mode active`	**Note** A port-channel must be in LACP Active mode for a vPC.

	Command or Action	Purpose
Step 7	**exit** **Example:** `apic1(config-if)# exit`	Returns to configure mode.
Step 8	**leaf** *node-id1 node-id2* **Example:** `apic1(config)# leaf 101-102`	Specifies the pair of leafs to be configured.
Step 9	**interface** *type* **Example:** `apic1(config-leaf)# interface ethernet 1/3-4`	Specifies the interface or range of interfaces that you are configuring to the port-channel.
Step 10	**[no] channel-group** *channel-name* **vpc** **Example:** `apic1(config-leaf-if)# channel-group foo vpc`	Assigns the interface or range of interfaces to the port-channel. Use the keyword **no** to remove the interface from the port-channel. To change the port-channel assignment on an interface, you can enter the **channel-group** command without first removing the interface from the previous port-channel. **Note** The **vpc** keyword in this command makes the port-channel a vPC. If the vPC does not already exist, a vPC ID is automatically generated and is applied to all member leaf nodes.
Step 11	**exit** **Example:** `apic1(config-leaf-if)# exit`	
Step 12	**exit** **Example:** `apic1(config-leaf)# exit`	
Step 13	**vpc context leaf** *node-id1 node-id2* **Example:** `apic1(config)# vpc context leaf 101 102`	The vpc context mode allows configuration of vPC to be applied to both leaf node pairs.
Step 14	**interface vpc** *channel-name* **Example:** `apic1(config-vpc)# interface vpc blue fex 102 102`	

	Command or Action	Purpose
Step 15	**[no] shutdown** **Example:** `apic1(config-vpc-if)# no shut`	(Optional) Administrative state configuration in the vpc context allows changing the admin state of a vPC with one command for both leaf nodes.

This example shows how to configure a basic vPC.

```
apic1(config)# vpc domain explicit 1 leaf 101 102
apic1(config)# template port-channel foo
apic1(config-if)# lacp mode active
apic1(config-if)# exit
apic1(config)# leaf 101-102
apic1(config-leaf)# interface ethernet 1/3-4
apic1(config-leaf-if)# channel-group foo vpc
apic1(config-leaf-if)# exit
```

This example shows how to configure vPCs with FEX ports.

```
apic1(config-leaf)# interface ethernet 101/1/1-2
apic1(config-leaf-if)# channel-group bar vpc
apic1(config)# vpc context leaf 101 102
apic1(config-vpc)# interface vpc foo
apic1(config-vpc-if)# exit
apic1(config-vpc)# interface vpc red fex 101 101
apic1(config-vpc-if)# switchport
apic1(config-vpc-if)# exit
apic1(config-vpc)# interface vpc blue fex 102 102
apic1(config-vpc-if)# shut
```

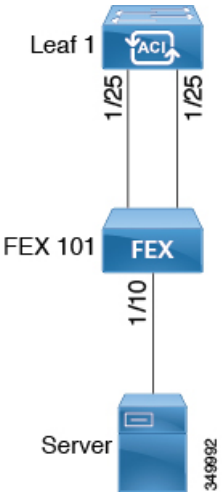
Basic FEX Configuration

The procedure below uses a Quick Start wizard that automatically creates some necessary policies for FEX deployment. The main steps are as follows:

1 Configure a switch profile that includes an auto-generated FEX profile.

2 Customize the auto-generated **FEX Profile** to enable attaching a server to a single FEX port.

Figure 14: Basic FEX Configuration



Note This procedure provides the steps for attaching a server to the FEX. The steps would be the same for attaching any device to an ACI attached FEX.

Before You Begin

- The ACI fabric is installed, APIC controllers are online, and the APIC cluster is formed and healthy.
- An APIC fabric administrator account is available that will enable creating the necessary fabric infrastructure configurations.
- The target leaf switch, interfaces, and protocol(s) are configured and available.
- The FEX is powered on and connected to the target leaf interfaces

Procedure

Step 1 On the APIC, create a switch profile using the **Fabric** > **Access Policies** > **Quick StartConfigure Interface, PC, And VPC** wizard.

a) On the APIC menu bar, navigate to **Fabric** > **Access Policies** > **Quick Start**.
b) In the **Quick Start** page, click the **Configure an interface, PC, and VPC** option to open the **Configure Interface, PC And VPC** wizard.
c) In the **Configure an interface, PC, and VPC** work area, click the + to add a new switch profile.
d) In the **Select Switches To Configure Interfaces** work area, click the **Advanced** radio button.
e) Select the switch. from the drop-down list of available switch IDs.

Troubleshooting Tips

In this procedure, one switch is included in the profile. Selecting multiple switches allows the same profile to be used on multiple switches.

f) Provide a name in the *Switch Profile Name* field.
g) Click the + above the Fexes list to add a FEX ID and the switch ports to which it will connect to the switch profile.
h) Click **Save** to save the changes. Click **Submit** to submit the switch proflle to the APIC.
The APIC auto-generates the necessary FEX profile (<*switch policy name*>_*FexP*<*FEX ID*>) and selector (<*switch policy name*>_*ifselctor*).

Verification: Use the CLI **show fex** command on the switch where the FEX is attached to verify that the FEX is online.

Step 2 Customize the auto-generated **FEX Profile** to enable attaching a server to a single FEX port.

a) In the **Navigation** pane, locate the switch policy you just created in the policies list. You will also find the auto-genterated FEX the <*switch policy name*>_*FexP*<*FEX ID*> profile.
b) In the work pane of the <*switch policy name*>_*FexP*<*FEX ID*> profile, click the + to add a new entry to the *Interface Selectors For FEX* list.
The **Create Access Port Selector** dialog opens.
c) Provide a name for the selector.
d) Specify the FEX interface IDs to use.
e) Select an existing *Interface Policy Group* from the list or *Create Access Port Policy Group*.
The access port policy group is a named policy that specifies the group of interface policies you will apply to the selected interfaces of the FEX. Examples of interface policies include Link Level Policy (for example, 1gbit port speed), Attach Entity Profile, Storm Control Interface Policy, and so forth.

Note Within the interface policy group, the *Attached Entity Profile* is required for enabling an EPG to use the interfaces specified in the FEX port selector.

f) Click **Submit** to submit the FEX proflle to the APIC.
The APIC updates the FEX profile.

Verification: Use the CLI **show int** command on the switch where the FEX is attached to verify that the FEX interface is configured accordingly.

What to Do Next

This completes the basic FEX configuration steps.

Note While this configuration enables hardware connectivity, no data traffic can flow without a valid application profile, EPG, and contract that is associated with this hardware configuration.

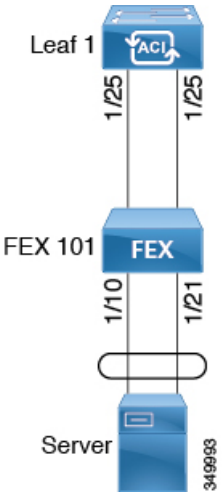
FEX Port Channel Configuration

The main steps are as follows:

1 Configure a FEX profile to use FEX ports to form a port channel.

2 Configure the port channel to enable attaching a server.

Figure 15: FEX port channel



Note This procedure provides the steps for attaching a server to the FEX port channel. The steps would be the same for attaching any device to an ACI attached FEX.

Before You Begin

- The ACI fabric is installed, APIC controllers are online, and the APIC cluster is formed and healthy.
- An APIC fabric administrator account is available that will enable creating the necessary fabric infrastructure configurations.
- The target leaf switch, interfaces, and protocol(s) are configured and available.
- The FEX is configured, powered on, and connected to the target leaf interfaces

Procedure

Step 1 On the APIC, add a port channel to a FEX profile.

a) On the APIC menu bar, navigate to **Fabric** > **Access Policies** > **Switch Policies** > **Profiles**.

b) In the **Navigation Pane**, select the FEX profile.
APIC auto-generated FEX profile names are formed as follows: *<switch policy name>_FexP<FEX ID>*.

c) In the **FEX Profile** work area, click the +to add a new entry to the *Interface Selectors For FEX* list.
The **Create Access Port Selector** dialog opens.

Step 2 Customize the **Create Access Port Selector** to enable attaching a server to the FEX port channel.

a) Provide a name for the selector.

b) Specify the FEX interface IDs to use.

c) Select an existing *Interface Policy Group* from the list or *Create PC Interface Policy Group*.

The port channel interface policy group specifies the group of policies you will apply to the selected interfaces of the FEX. Examples of interface policies include Link Level Policy (for example, 1gbit port speed), Attach Entity Profile, Storm Control Interface Policy, and so forth.

Note Within the interface policy group, the *Attached Entity Profile* is required for enabling an EPG to use the interfaces specified in the FEX port selector.

d) In the *Port Channel Policy* option, select static or dynamic LACP according to the requirements of your configuration.

e) Click **Submit** to submit the updated FEX proflle to the APIC.
The APIC updates the FEX profile.

Verification: Use the CLI **show port-channel summary** command on the switch where the FEX is attached to verify that the port channel is configured accordingly.

What to Do Next

This completes the FEX port channel configuration steps.

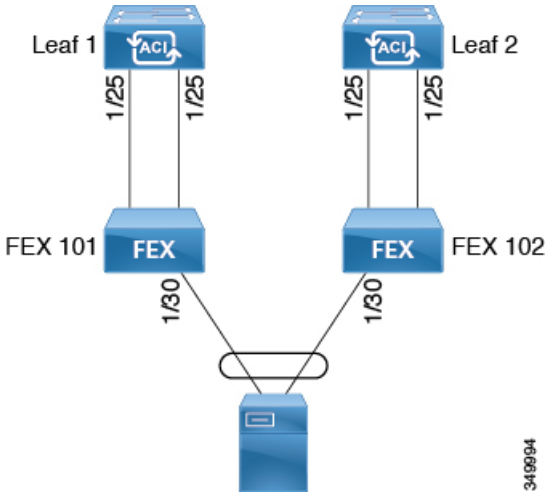
Note While this configuration enables hardware connectivity, no data traffic can flow without a valid application profile, EPG, and contract that is associated with this hardware configuration.

FEX Virtual Port Channel Configuration

The main steps are as follows:

1. Configure two existing FEX profiles to form a virtual port channel.
2. Configure the virtual port channel to enable attaching a server to the FEX port channel.

Figure 16: FEX virtual port channel



Note This procedure provides the steps for attaching a server to the FEX virtual port channel. The steps would be the same for attaching any device to an ACI attached FEX.

Before You Begin

- The ACI fabric is installed, APIC controllers are online, and the APIC cluster is formed and healthy.
- An APIC fabric administrator account is available that will enable creating the necessary fabric infrastructure configurations.
- The target leaf switch, interfaces, and protocol(s) are configured and available.
- The FEXes are configured, powered on, and connected to the target leaf interfaces

Procedure

Step 1 On the APIC, add a virtual port channel to two FEX profiles.

a) On the APIC menu bar, navigate to **Fabric** > **Access Policies** > **Switch Policies** > **Profiles**.

b) In the **Navigation Pane**, select the first FEX profile.
APIC auto-generated FEX profile names are formed as follows: <*switch policy name*>_*FexP*<*FEX ID*>.

c) In the **FEX Profile** work area, click the +to add a new entry to the *Interface Selectors For FEX* list.
The **Create Access Port Selector** dialog opens.

Step 2 Customize the **Create Access Port Selector** to enable attaching a server to the FEX virtual port channel.

a) Provide a name for the selector.

b) Specify the FEX interface ID to use.
Typically, you will use the same interface ID on each FEX to form the virtual port channel.

c) Select an existing *Interface Policy Group* from the list or *Create VPC Interface Policy Group*.
The virtual port channel interface policy group specifies the group of policies you will apply to the selected interfaces of the FEX. Examples of interface policies include Link Level Policy (for example, 1gbit port speed), Attach Entity Profile, Storm Control Interface Policy, and so forth.

Note Within the interface policy group, the *Attached Entity Profile* is required for enabling an EPG to use the interfaces specified in the FEX port selector.

d) In the *Port Channel Policy* option, select static or dynamic LACP according to the requirements of your configuration.

e) Click **Submit** to submit the updated FEX proflle to the APIC.
The APIC updates the FEX profile.

Verification: Use the CLI **show port-channel summary** command on the switch where the FEX is attached to verify that the port channel is configured accordingly.

Step 3 Configure the second FEX to use the same *Interface Policy Group* just specified for the first FEX.

a) In the **FEX Profile** work area of the second FEX profile, click the +to add a new entry to the *Interface Selectors For FEX* list.
The **Create Access Port Selector** dialog opens.

b) Provide a name for the selector.

c) Specify the FEX interface ID to use.
Typically, you will use the same interface ID on each FEX to form the virtual port channel.

d) From the drop-down list, select the same virtual port channel *Interface Policy Group* just used in the first FEX profile.
The virtual port channel interface policy group specifies the group of policies you will apply to the selected interfaces of the FEX. Examples of interface policies include Link Level Policy (for example, 1gbit port speed), Attach Entity Profile, Storm Control Interface Policy, and so forth.

Note Within the interface policy group, the *Attached Entity Profile* is required for enabling an EPG to use the interfaces specified in the FEX port selector.

e) Click **Submit** to submit the updated FEX proflle to the APIC.
The APIC updates the FEX profile.

Verification: Use the CLI **show vpc extended** command on the switch where one of the FEXes is attached to verify that the virtual port channel is configured accordingly.

What to Do Next

This completes the FEX virtual port channel configuration steps.

Note While this configuration enables hardware connectivity, no data traffic can flow without a valid application profile, EPG, and contract that is associated with this hardware configuration.

About Traffic Storm Control

A traffic storm occurs when packets flood the LAN, creating excessive traffic and degrading network performance. You can use traffic storm control policies to prevent disruptions on Layer 2 ports by broadcast, unknown multicast, or unknown unicast traffic storms on physical interfaces.

By default, storm control is not enabled in the ACI fabric. ACI bridge domain (BD) Layer 2 unknown unicast flooding is enabled by default within the BD but can be disabled by an administrator. In that case, a storm control policy only applies to broadcast and unknown multicast traffic. If Layer 2 unknown unicast flooding is enabled in a BD, then a storm control policy applies to Layer 2 unknown unicast flooding in addition to broadcast and unknown multicast traffic.

Traffic storm control (also called traffic suppression) allows you to monitor the levels of incoming broadcast, multicast, and unknown unicast traffic over a one second interval. During this interval, the traffic level, which is expressed either as percentage of the total available bandwidth of the port or as the maximum packets per second allowed on the given port, is compared with the traffic storm control level that you configured. When the ingress traffic reaches the traffic storm control level that is configured on the port, traffic storm control drops the traffic until the interval ends. An administrator can configure a monitoring policy to raise a fault when a storm control threshold is exceeded.

Storm Control Guidelines

Configure traffic storm control levels according to the following guidelines and limitations:

- Typically, a fabric administrator configures storm control in fabric access policies on the following interfaces:

- A regular trunk interface.
- A direct port channel on a single leaf switch.
- A virtual port channel (a port channel on two leaf switches).

- For port channels and virtual port channels, the storm control values (packets per second or percentage) apply to all individual members of the port channel. Do not configure storm control on interfaces that are members of a port channel.

Note On switch hardware starting with the APIC 1.3(x) and switch 11.3(x) release, for port channel configurations, the traffic suppression on the aggregated port may be up to two times the configured value. The new hardware ports are internally subdivided into these two groups: slice-0 and slice-1. To check the slicing map, use the `vsh_lc` command `show platform internal hal l2 port gpd` and look for `slice 0` or `slice 1` under the `Sl` column. If port-channel members fall on both slice-0 and slice-1, allowed storm control traffic may become twice the configured value because the formula is calculated based on each slice.

- When configuring by percentage of available bandwidth, a value of 100 means no traffic storm control and a value of 0.01 suppresses all traffic.
- Due to hardware limitations and the method by which packets of different sizes are counted, the level percentage is an approximation. Depending on the sizes of the frames that make up the incoming traffic, the actual enforced level might differ from the configured level by several percentage points. Packets-per-second (PPS) values are converted to percentage based on 256 bytes.
- Maximum burst is the maximum accumulation of rate that is allowed when no traffic passes. When traffic starts, all the traffic up to the accumulated rate is allowed in the first interval. In subsequent intervals, traffic is allowed only up to the configured rate. The maximum supported is 65535 KB. If the configured rate exceeds this value, it is capped at this value for both PPS and percentage.
- The maximum burst that can be accumulated is 512 MB.
- On an egress leaf switch in optimized multicast flooding (OMF) mode, traffic storm control will not be applied.
- On an egress leaf switch in non-OMF mode, traffic storm control will be applied.
- On a leaf switch for FEX, traffic storm control is not available on host-facing interfaces.

Configuring a Traffic Storm Control Policy Using the GUI

Procedure

Step 1 In the menu bar, click **Fabric**.

Step 2 In the submenu bar, click **Access Policies**.

Step 3 In the **Navigation** pane, expand **Interface Policies**.

Step 4 Expand **Policies**.

Step 5 Right-click **Storm Control** and choose **Create Storm Control Interface Policy**.

Step 6 In the **Create Storm Control Interface Policy** dialog box, enter a name for the policy in the **Name** field.

Step 7 In the **Specify Policy In** field, click the radio button for either **Percentage** or **Packets Per Second**.

Step 8 If you chose **Percentage**, perform the following steps:

a) In the **Rate** field, enter a traffic rate percentage.
Enter a number between 0 and 100 that specifies a percentage of the total available bandwidth of the port. When the ingress traffic reaches this level during a one second interval, traffic storm control drops traffic for the remainder of the interval. A value of 100 means no traffic storm control. A value of 0 suppresses all traffic.

b) In the **Max Burst Rate** field, enter a burst traffic rate percentage.
Enter a number between 0 and 100 that specifies a percentage of the total available bandwidth of the port. When the ingress traffic reaches this level, traffic storm control begins to drop traffic.

Step 9 If you chose **Packets Per Second**, perform the following steps:

a) In the **Rate** field, enter a traffic rate in packets per second.
During this interval, the traffic level, expressed as packets flowing per second through the port, is compared with the traffic storm control level that you configured. When the ingress traffic reaches the traffic storm control level that is configured on the port, traffic storm control drops the traffic until the interval ends.

b) In the **Max Burst Rate** field, enter a burst traffic rate in packets per second.
During this interval, the traffic level, expressed as packets flowing per second through the port, is compared with the burst traffic storm control level that you configured. When the ingress traffic reaches the traffic storm control level that is configured on the port, traffic storm control drops the traffic until the interval ends.

Step 10 Click **Submit**.

Step 11 Apply the storm control interface policy to an interface port.

a) In the menu bar, click **Fabric**.
b) In the submenu bar, click **Access Policies**.
c) In the **Navigation** pane, expand **Interface Policies**.
d) Expand **Policy Groups**.
e) Select **Policy Group**.
f) In the **Work** pane, click the drop down for **Storm Control Interface Policy** and select the created **Traffic Storm Control Policy**.
g) Click **Submit**.

Configuring a Traffic Storm Control Policy Using the REST API

To configure a traffic storm control policy, create a `stormctrl:IfPol` object with the desired properties.

To create a policy named MyStormPolicy, send this HTTP POST message:

```
POST https://192.0.20.123/api/mo/uni/infra/stormctrlifp-MyStormPolicy.json
```

In the body of the POST message, Include the following JSON payload structure to specify the policy by percentage of available bandwidth:

```
{"stormctrlIfPol":
    {"attributes":
        {"dn":"uni/infra/stormctrlifp-MyStormPolicy",
        "name":"MyStormPolicy",
        "rate":"75",
        "burstRate":"85",
        "rn":"stormctrlifp-MyStormPolicy",
        "status":"created"
        },
        "children":[]
    }
}
```

In the body of the POST message, Include the following JSON payload structure to specify the policy by packets per second:

```
{"stormctrlIfPol":
    {"attributes":
        {"dn":"uni/infra/stormctrlifp-MyStormPolicy",
        "name":"MyStormPolicy",
        "ratePps":"12000",
        "burstPps":"15000",
        "rn":"stormctrlifp-MyStormPolicy",
        "status":"created"
        },
        "children":[]
    }
}
```

Apply the traffic storm control interface policy to an interface port.

```
POST
http://192.0.20.123/api/node/mo/uni/infra/funcprof/accportgrp-InterfacePolicyGroup/rsstormctrlIfPol.json
```

In the body of the POST message, Include the following JSON payload structure to apply the policy to the interface policy group.

```
{"infraRsStormctrlIfPol":{"attributes":{"tnStormctrlIfPolName":"testStormControl"},"children":[]}}
```

Configuring a Traffic Storm Control Policy Using the NX-OS Like CLI

Procedure

	Command or Action	Purpose

Step 1	Enter the following commands to create a PPS policy:
	
	Example:
	`(config)# template policy-group pg1`
	`(config-pol-grp-if)# storm-control pps 10000 burst-rate 10000`	

	Command or Action	Purpose
Step 2	Enter the following commands to create a percent policy:	

```
(config)# template policy-group pg2
(config-pol-grp-if)# storm-control level 50 burst-rate 60
```

Intra-EPG Endpoint Isolation

Intra-EPG endpoint isolation policies provide full isolation for virtual or physical endpoints; no communication is allowed between endpoints in an EPG that is operating with isolation enforced. Isolation enforced EPGs reduce the number of EPG encapsulations required when many clients access a common service but are not allowed to communicate with each other.

An EPG is isolation enforced for all ACI network domains or none. While the ACI fabric implements isolation directly to connected endpoints, switches connected to the fabric are made aware of isolation rules according to a primary VLAN (PVLAN) tag.

Note If an EPG is configured with intra-EPG endpoint isolation enforced, these restrictions apply:

- All Layer 2 endpoint communication across an isolation enforced EPG is dropped within a bridge domain.
- All Layer 3 endpoint communication across an isolation enforced EPG is dropped within the same subnet.

Intra-EPG Isolation for Bare Metal Servers

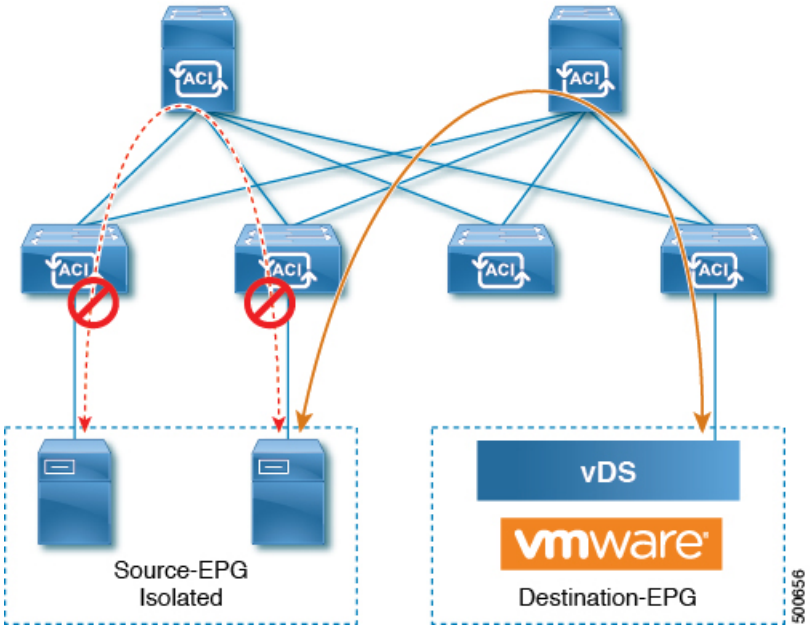
Intra-EPG endpoint isolation policies can be applied to directly connected endpoints such as bare metal servers.

Examples use cases include the following:

- Backup clients have the same communication requirements for accessing the backup service, buy they don't need to communicate with each other.

- Servers behind a load balancer have the same communication requirements, but isolating them from each other protects against a server that is compromised or infected.

Figure 17: Intra-EPG Isolation for Bare Metal Servers



Bare metal EPG isolation is enforced at the leaf switch. Bare metal servers use VLAN encapsulation. All unicast, multicast and broadcast traffic is dropped (denied) within isolation enforced EPGs. ACI bridge-domains can have a mix of isolated and regular EPGs. Each Isolated EPG can have multiple VLANs where intra-vlan traffic is denied.

Using the GUI to Configure Intra-EPG Isolation for Bare Metal Servers

The port the EPG uses must be associated with a bare metal server interface in the physical domain that is used to connect the bare metal servers directly to leaf switches.

Procedure

Step 1 In a tenant, right click on an **Application Profile**, and open the **Create Application EPG** dialog box to perform the following actions:

a) In the **Name** field, add the EPG name (intra_EPG-deny).
b) For **Intra EPG Isolation**, click **Enforced**.
c) In the **Bridge Domain** field, choose the bridge domain from the drop-down list (bd1).
d) Check the **Statically Link with Leaves/Paths** check box.
e) Click **Next**.

Step 2 In the **Leaves/Paths** dialog box, perform the following actions:

a) In the **Path** section, choose a path from the drop-down list (Node-107/eth1/16) in Trunk Mode. Specify the **Port Encap** (vlan-102) for the secondary VLAN.

Note If the bare metal server is directly connected to a leaf switch, only the Port Encab secondary VLAN is specified.

Specify the **Primary Encap** (vlan-103) for the primary VLAN.

b) Click **Update**.

c) Click **Finish**.

Using the NX-OS Style CLI to Configure Intra-EPG Isolation for Bare Metal Servers

Procedure

	Command or Action	Purpose

Step 1	In the CLI, create an intra-EPG isolation EPG:
	
	Example:
	The VMM case is below.
	`ifav19-ifc1(config)# tenant Test_Isolation`
	`ifav19-ifc1(config-tenant)# application PVLAN`
	`ifav19-ifc1(config-tenant-app)# epg EPG1`
	`ifav19-ifc1(config-tenant-app-epg)# show running-config`
	`# Command: show running-config`
	`tenant Test_Isolation`
	`application PVLAN epg EPG1`
	`tenant Test_Isolation`
	`application PVLAN`
	`epg EPG1`
	`bridge-domain member BD1`
	`contract consumer bare-metal`
	`contract consumer default`
	`contract provider Isolate_EPG`
	`isolation enforce <---- This enables EPG isolation mode.`
	`exit`
	`exit`
	`ifav19-ifc1(config)# leaf ifav19-leaf3`
	`ifav19-ifc1(config-leaf)# interface ethernet 1/16`
	`ifav19-ifc1(config-leaf-if)# show running-config`
	`ifav19-ifc1(config-leaf-if)# switchport trunk native vlan 101 tenant Test_Isolation application PVLAN epg StaticEPG primary-vlan 100`
	`exit`	

Step 2	Verify the configuration:
	
	Example:
	`show epg StaticEPG detail`
	`Application EPg Data:`
	`Tenant : Test_Isolation`
	`Application : PVLAN`
	`AEPg : StaticEPG`
	`BD : BD1`
	`uSeg EPG : no`
	`Intra EPG Isolation : enforced`
	`Vlan Domains : phys`
	`Consumed Contracts : bare-metal`
	`Provided Contracts : default,Isolate_EPG`
	`Denied Contracts :`
	`Qos Class : unspecified`
	`Tag List :`
	`VMM Domains:`	

	Command or Action	Purpose

```
Domain                Type       Deployment Immediacy  Resolution Immediacy
 State           Encap       Primary
Encap
 --------------------  ---------  --------------------  --------------------
  --------------  ----------  ----------
 DVS1                 VMware     On Demand             immediate
  formed          auto        auto

Static Leaves:
 Node        Encap             Deployment Immediacy  Mode
Modification Time
----------  ----------------  --------------------  ------------------
------------------------------

Static Paths:
 Node        Interface                        Encap             Modification
Time
 ----------  -------------------------------  ----------------
------------------------------
 1018        eth101/1/1                       vlan-100
2016-02-11T18:39:02.337-08:00
 1019        eth1/16                          vlan-101
2016-02-11T18:39:02.337-08:00

Static Endpoints:
 Node        Interface          Encap            End Point MAC      End Point
 IP Address          Modification Time
 ----------  -----------------------------  ----------------
-----------------  ------------------------------
------------------------------
```

Using the REST API to Configure Intra-EPG Isolation for Bare Metal Servers

Before You Begin

The port the EPG uses must be associated with a bare metal server interface in the physical domain.

Procedure

Step 1 Send this HTTP POST message to deploy the application using the XML API.

Example:

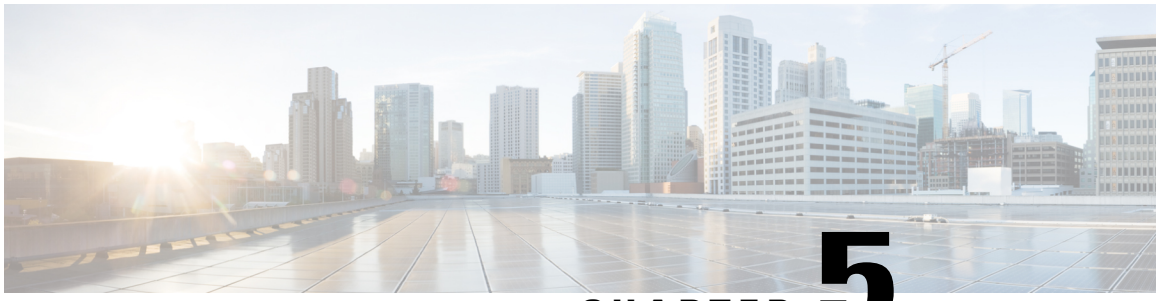
```
   POST
https://192.0.20.123/api/mo/uni/tn-ExampleCorp.xml
```

Step 2 Include this XML structure in the body of the POST message.

Example:

```
<fvTenant name="Tenant_BareMetal" >
  <fvAp name="Web">
    <fvAEPg name="IntraEPGDeny" pcEnfPref="enforced">
          <!-- pcEnfPref="enforced" ENABLES ISOLATION-->
          <fvRsBd tnFvBDName="bd" />
          <fvRsDomAtt tDn="uni/phys-Dom1" />
          <!-- PATH ASSOCIATION -->
          <fvRsPathAtt tDn="topology/pod-1/paths-1017/pathep-[eth1/2]" encap="vlan-51"
```

```
primaryEncap="vlan-100" instrImedcy='immediate'/>
    </fvAEPg>
 </fvAp>
</fvTenant>
```



CHAPTER 5

Basic User Tenant Configuration

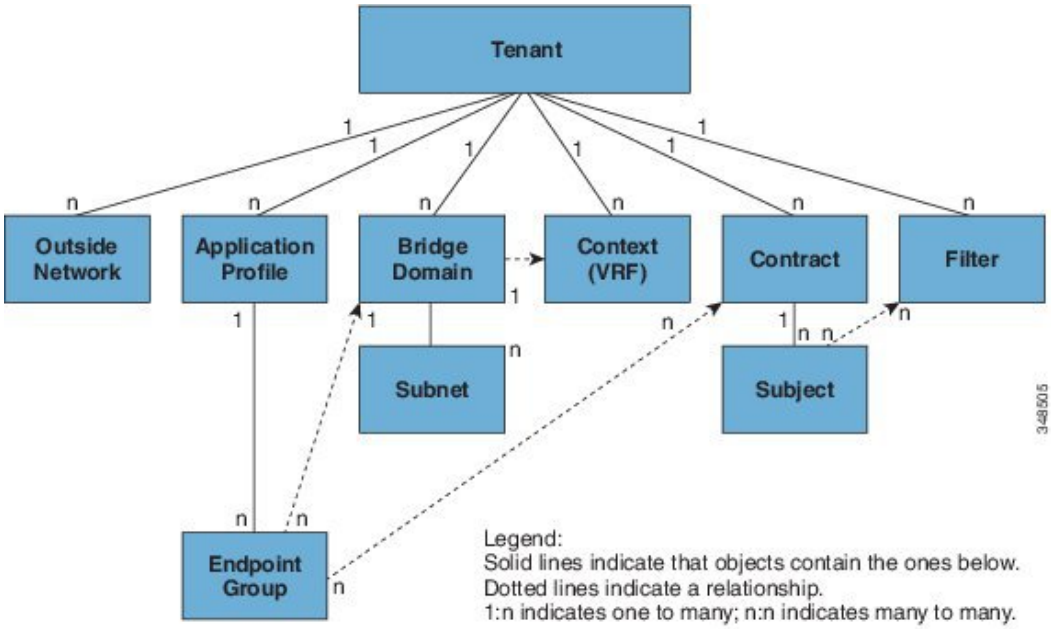
This chapter contains the following sections:

Tenants

A tenant (`fvTenant`) is a logical container for application policies that enable an administrator to exercise domain-based access control. A tenant represents a unit of isolation from a policy perspective, but it does not represent a private network. Tenants can represent a customer in a service provider setting, an organization

or domain in an enterprise setting, or just a convenient grouping of policies. The following figure provides an overview of the tenant portion of the management information tree (MIT).

Figure 18: Tenants



Tenants can be isolated from one another or can share resources. The primary elements that the tenant contains are filters, contracts, outside networks, bridge domains, contexts, and application profiles that contain endpoint groups (EPGs). Entities in the tenant inherit its policies. A tenant can contain one or more virtual routing and forwarding (VRF) instances or contexts; each context can be associated with multiple bridge domains.

Note In the APIC GUI under the tenant navigation path, a context (VRF) is called a private network.

Tenants are logical containers for application policies. The fabric can contain multiple tenants. You must configure a tenant before you can deploy any Layer 4 to Layer 7 services. The ACI fabric supports IPv4, IPv6, and dual-stack configurations for tenant networking.

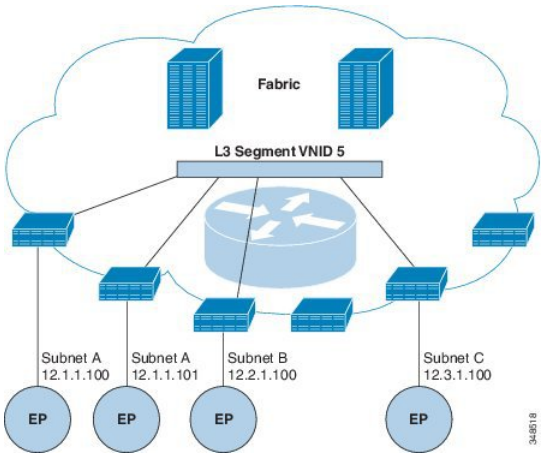
Routing Within the Tenant

The Application Centric Infrastructure (ACI) fabric provide tenant default gateway functionality and route between the fabric virtual extensible local area (VXLAN) networks. For each tenant, the fabric provides a virtual default gateway or Switched Virtual Interface (SVI) whenever a subnet is created on the APIC. This spans any switch that has a connected endpoint for that tenant subnet. Each ingress interface supports the default gateway interface and all of the ingress interfaces across the fabric share the same router IP address and MAC address for a given tenant subnet.

Layer 3 VNIDs Used to Transport Intersubnet Tenant Traffic

In the ACI model, traffic that arrives at the fabric ingress that is sent to the ACI fabric default gateway is routed into a virtual network segment known as the Layer 3 VNID. A single Layer 3 VNID is assigned for each tenant context. The following figure shows how routing within the tenant is done.

Figure 19: Layer 3 VNIDs Transport Intersubnet Tenant Traffic



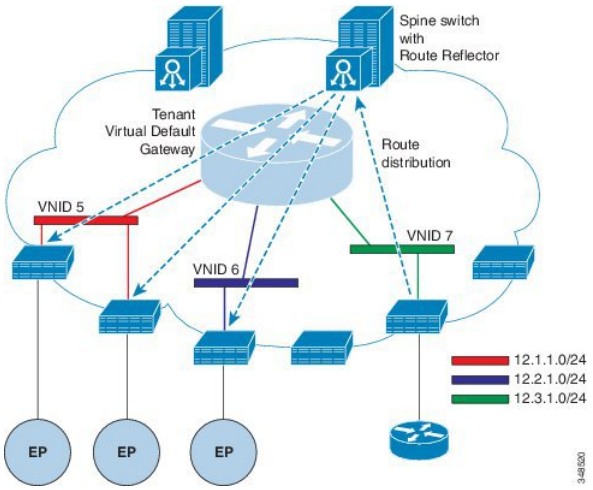
The Layer 3 VNID is allocated by the APIC. The traffic that goes across the fabric is transported using the VNID of the Layer 3 segment. In the egress leaf switch, the packet is routed from the Layer 3 segment VNID to the VNID of the egress subnet.

The ACI model provides very efficient forwarding in the fabric for the traffic that is routed within the tenant. For example, the traffic between two virtual machines (VM) that belongs to the same tenant on the same physical host, but on different subnets travels only to the ingress switch before it is routed (using the minimal path cost) to the correct destination. In the current VM environments, the traffic travels to an edge VM (possibly on a different physical server) before it is routed to the correct destination.

Router Peering and Route Distribution

As shown in the figure below, when the routing peer model is used, the leaf switch interface is statically configured to peer with the external router's routing protocol.

Figure 20: Router Peering

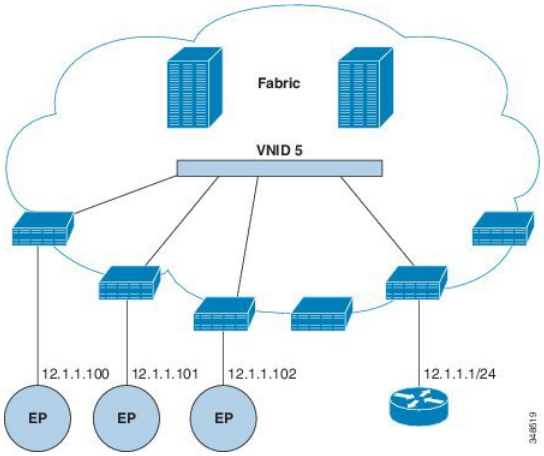


The routes that are learned through peering are sent to the spine switches. The spine switches act as route reflectors and distribute the external routes to all of the leaf switches that have interfaces that belong to the same tenant. These routes are longest prefix match (LPM) summarized addresses and are placed in the leaf switch's forwarding table with the VTEP IP address of the remote leaf switch where the external router is connected. WAN routes have no forwarding proxy. If the WAN routes do not fit in the leaf switch's forwarding table, the traffic is dropped. Because the external router is not the default gateway, packets from the tenant endpoints (EPs) are sent to the default gateway in the ACI fabric.

Bridged Interface to an External Router

As shown in the figure below, when the leaf switch interface is configured as a bridged interface, the default gateway for the tenant VNID is the external router.

Figure 21: Bridged External Router



The ACI fabric is unaware of the presence of the external router and the APIC statically assigns the leaf switch interface to its EPG.

Configuring Route Reflectors

The ACI fabric route reflectors use multiprotocol BGP (MP-BGP) to distribute external routes within the fabric. To enable route reflectors in the ACI fabric, the fabric administrator must select the spine switches that will be the route reflectors, and provide the autonomous system (AS) number. Once route reflectors are enabled in the ACI fabric, administrators can configure connectivity to external networks as described in the following sections.

To connect external routers to the ACI fabric, the fabric infrastructure administrator configures spine nodes as Border Gateway Protocol (BGP) route reflectors. For redundancy purposes, more than one spine is configured as a router reflector node (one primary and one secondary reflector).

When a tenant needs to attach a WAN router to the ACI fabric, the infrastructure administrator configures the leaf node (as described below) to which the WAN router is being connected as WAN top of rack (ToR) and pairs this WAN ToR with one of the route reflector nodes as a BGP peer. When route reflectors are configured on the WAN ToR, they are able to advertise the tenant routes into the fabric.

Each leaf node can store up to 4000 routes. If a WAN router has to advertise more than 4000 routes, it should peer with multiple leaf nodes. The infrastructure administrator configures each of the paired leaf nodes with the routes (or route prefixes) that it can advertise.

The infrastructure administrator must configure an external WAN router connected to the fabric as follows:

1. Configure up to two spine nodes as route reflectors. For redundancy, configure primary and secondary route reflectors.
2. On WAN ToRs, configure the primary and secondary route reflector nodes.

3 On WAN ToRs, configure the routes that the ToR is responsible for advertising. This is optional and needs to be done only when the tenant router is known to advertise more than 4000 routes.

Configuring External Connectivity for Tenants

Before you can distribute the static route to the other leaf switches on the Application Centric Infrastructure (ACI) fabric, a multiprotocol BGP (MP-BGP) process must first be operating, and the spine switches must be configured as BGP route reflectors.

To integrate the ACI fabric into an external routed network, you can configure Open Shortest Path First (OSPF) for management tenant Layer 3 connectivity.

Configuring an MP-BGP Route Reflector Using the Advanced GUI

Note To watch an example video of this task, see Videos Webpage.

Procedure

Step 1 On the menu bar, choose **FABRIC** > **Fabric Policies**.

Step 2 In the **Navigation** pane, expand **Pod Policies** > **Policies** > **BGP Route Reflector default**, right-click **BGP Route Reflector default**, and click **Create Route Reflector Node Policy EP**.

Step 3 In the **Create Route Reflector Node Policy EP** dialog box, from the **Spine Node** drop-down list, choose the appropriate spine node. Click **Submit**.

Note Repeat the above steps to add additional spine nodes as required.

The spine switch is marked as the route reflector node.

Step 4 In the **BGP Route Reflector default** properties area, in the **Autonomous System Number** field, choose the appropriate number. Click **Submit**.

Note The autonomous system number must match the leaf connected router configuration if Border Gateway Protocol (BGP) is configured on the router. If you are using routes learned using static or Open Shortest Path First (OSPF), the autonomous system number value can be any valid value.

Step 5 In the **Navigation** pane, expand and right-click **Policy Groups**, and click **Create POD Policy Group**.

Step 6 In the **Create POD Policy Group** dialog box, in the **Name** field, enter the name of a pod policy group.

Step 7 In the **BGP Route Reflector Policy** drop-down list, choose the appropriate policy (default). Click **Submit**.
The BGP route reflector policy is associated with the route reflector pod policy group, and the BGP process is enabled on the leaf switches.

Step 8 In the **Navigation** pane, choose **Pod Policies** > **Profiles** > **default**. In the **Work** pane, from the **Fabric Policy Group** drop-down list, choose the pod policy that was created earlier. Click **Submit**.
The pod policy group is now applied to the fabric policy group.

Creating an OSPF External Routed Network for Management Tenant Using the Advanced GUI

- You must verify that the router ID and the logical interface profile IP address are different and do not overlap.
- The following steps are for creating an OSPF external routed network for a management tenant. To create an OSPF external routed network for a tenant, you must choose a tenant and create a VRF for the tenant.
- For more details, see also the KB article about *Transit Routing*.

Note To watch an example video of this task, see Videos Webpage.

Procedure

Step 1 On the menu bar, choose **TENANTS** > **mgmt**.

Step 2 In the **Navigation** pane, expand **Networking** > **External Routed Networks**.

Step 3 Right-click **External Routed Networks**, and click **Create Routed Outside**.

Step 4 In the **Create Routed Outside** dialog box, perform the following actions:

a) In the **Name** field, enter a name (RtdOut).
b) Check the **OSPF** check box.
c) In the **OSPF Area ID** field, enter an area ID.
d) In the **OSPF Area Control** field, check the appropriate check box.
e) In the **OSPF Area Type** field, choose the appropriate area type.
f) In the **OSPF Area Cost** field, choose the appropriate value.
g) In the **VRF** field, from the drop-down list, choose the VRF (inb).
Note This step associates the routed outside with the in-band VRF.
h) From the **External Routed Domain** drop-down list, choose the appropriate domain.
i) Click the + icon for **Nodes and Interfaces Protocol Profiles** area.

Step 5 In the **Create Node Profile** dialog box, perform the following actions:

a) In the **Name** field, enter a name for the node profile. (borderLeaf).
b) In the **Nodes** field, click the + icon to display the **Select Node** dialog box.
c) In the **Node ID** field, from the drop-down list, choose the first node. (leaf1).
d) In the **Router ID** field, enter a unique router ID.
e) Uncheck the **Use Router ID as Loopback Address** field.
Note By default, the router ID is used as a loopback address. If you want them to be different, uncheck the **Use Router ID as Loopback Address** check box.
f) Expand **Loopback Addresses**, and enter the IP address in the **IP** field. Click **Update**, and click **OK**. Enter the desired IPv4 or IPv6 IP address.
g) In the **Nodes** field, expand the + icon to display the **Select Node** dialog box.
Note You are adding a second node ID.
h) In the **Node ID** field, from the drop-down list, choose the next node. (leaf2).

i) In the **Router ID** field, enter a unique router ID.
j) Uncheck the **Use Router ID as Loopback Address** field.
Note By default, the router ID is used as a loopback address. If you want them to be different, uncheck the **Use Router ID as Loopback Address** check box.
k) Expand **Loopback Addresses**, and enter the IP address in the **IP** field. Click **Update**, and click **OK**. Click **OK**.
Enter the desired IPv4 or IPv6 IP address.

Step 6 In the **Create Node Profile** dialog box, in the **OSPF Interface Profiles** area, click the + icon.

Step 7 In the **Create Interface Profile** dialog box, perform the following tasks:
a) In the **Name** field, enter the name of the profile (portProf).
b) In the **Interfaces** area, click the **Routed Interfaces** tab, and click the + icon.
c) In the **Select Routed Interfaces** dialog box, in the **Path** field, from the drop-down list, choose the first port (leaf1, port 1/40).
d) In the **IP Address** field, enter an IP address and mask. Click **OK**.
e) In the **Interfaces** area, click the **Routed Interfaces** tab, and click the + icon.
f) In the **Select Routed Interfaces** dialog box, in the **Path** field, from the drop-down list, choose the second port (leaf2, port 1/40).
g) In the **IP Address** field, enter an IP address and mask. Click **OK**.
Note This IP address should be different from the IP address you entered for leaf1 earlier.
h) In the **Create Interface Profile** dialog box, click **OK**.
The interfaces are configured along with the OSPF interface.

Step 8 In the **Create Node Profile** dialog box, click **OK**.

Step 9 In the **Create Routed Outside** dialog box, click **Next**.
The **Step 2 External EPG Networks** area is displayed.

Step 10 In the **External EPG Networks** area, click the + icon.

Step 11 In the **Create External Network** dialog box, perform the following actions:
a) In the **Name** field, enter a name for the external network (extMgmt).
b) Expand **Subnet** and in the **Create Subnet** dialog box, in the **IP address** field, enter an IP address and mask for the subnet.
c) In the **Scope** field, check the desired check boxes. Click **OK**.
d) In the **Create External Network** dialog box, click **OK**.
e) In the **Create Routed Outside** dialog box, click **Finish**.
Note In the **Work** pane, in the **External Routed Networks** area, the external routed network icon (RtdOut) is now displayed.

Configuring an MP-BGP Route Reflector Using the REST API

Procedure

Step 1 Mark the spine switches as route reflectors.

Example:

```
POST URL: https://apic-ip/api/policymgr/mo/uni/fabric.xml

<bgpInstPol name="default">
  <bgpAsP asn="1" />
  <bgpRRP>
     <bgpRRNodePEp id="<spine_id1>"/>
     <bgpRRNodePEp id="<spine_id2>"/>
  </bgpRRP>
</bgpInstPol>
```

Step 2 Set up the pod selector using the following post.

Example:

For the FuncP setup—

```
POST URL:
https://APIC-IP/api/policymgr/mo/uni.xml

<fabricFuncP>
  <fabricPodPGrp name="bgpRRPodGrp">
    <fabricRsPodPGrpBGPRRP tnBgpInstPolName="default" />
  </fabricPodPGrp>
</fabricFuncP>
```

Example:

For the PodP setup—

```
POST URL:
https://APIC-IP/api/policymgr/mo/uni.xml

<fabricPodP name="default">
  <fabricPodS name="default" type="ALL">
    <fabricRsPodPGrp tDn="uni/fabric/funcprof/podpgrp-bgpRRPodGrp"/>
  </fabricPodS>
</fabricPodP>
```

Verifying the MP-BGP Route Reflector Configuration

Procedure

Step 1 Verify the configuration by performing the following actions:

a) Use secure shell (SSH) to log in as an administrator to each leaf switch as required.
b) Enter the **show processes | grep bgp** command to verify the state is S.
If the state is NR (not running), the configuration was not successful.

Step 2 Verify that the autonomous system number is configured in the spine switches by performing the following actions:

a) Use the SSH to log in as an administrator to each spine switch as required.
b) Execute the following commands from the shell window

Example:

cd /mit/sys/bgp/inst

Example:
grep asn summary

The configured autonomous system number must be displayed. If the autonomous system number value displays as 0, the configuration was not successful.

Creating Tenants, VRF, and Bridge Domains

Tenants Overview

- A tenant contains policies that enable qualified users domain-based access control. Qualified users can access privileges such as tenant administration and networking administration.
- A user requires read/write privileges for accessing and configuring policies in a domain. A tenant user can have specific privileges into one or more domains.
- In a multitenancy environment, a tenant provides group user access privileges so that resources are isolated from one another (such as for endpoint groups and networking). These privileges also enable different users to manage different tenants.

Tenant Creation

A tenant contains primary elements such as filters, contracts, bridge domains, and application profiles that you can create after you first create a tenant.

VRF and Bridge Domains

You can create and specify a VRF and a bridge domain for the tenant. The defined bridge domain element subnets reference a corresponding Layer 3 context.

For details about enabling IPv6 Neighbor Discovery see the related KB article, *KB: Creating a Tenant, VRF, and Bridge Domain with IPv6 Neighbor Discovery* .

Creating a Tenant, VRF, and Bridge Domain Using the Advanced GUI

- To watch an example video of this task, see Videos Webpage.
- If you have a public subnet when you configure the routed outside, you must associate the bridge domain with the outside configuration.

Procedure

Step 1 On the menu bar, click **TENANT** > **Add Tenant**.

Step 2 In the **Create Tenant** dialog box, perform the following tasks:

a) In the **Name** field, enter a name.
b) Click the **Security Domains** + icon to open the **Create Security Domain** dialog box.
c) In the **Name** field, enter a name for the security domain. Click **Submit**.
d) In the **Create Tenant** dialog box, check the check box for the security domain that you created, and click **Submit**.

Step 3 In the **Navigation** pane, expand **Tenant-name** > **Networking**, and in the **Work** pane, drag the **VRF** icon to the canvas to open the **Create VRF** dialog box, and perform the following tasks:

a) In the **Name** field, enter a name.
b) Click **Submit** to complete the VRF configuration.

Step 4 In the **Networking** pane, drag the **BD** icon to the canvas while connecting it to the **VRF** icon. In the **Create Bridge Domain** dialog box that displays, perform the following tasks:

a) In the **Name** field, enter a name.
b) Click the **L3 Configurations** tab.
c) Expand **Subnets** to open the **Create Subnet** dialog box, enter the subnet mask in the **Gateway IP** field and click **OK**.
d) Click **Submit** to complete bridge domain configuration.

Step 5 In the **Networks** pane, drag the **L3** icon down to the canvas while connecting it to the **VRF** icon. In the **Create Routed Outside** dialog box that displays, perform the following tasks:

a) In the **Name** field, enter a name.
b) Expand **Nodes And Interfaces Protocol Profiles** to open the **Create Node Profile** dialog box.
c) In the **Name** field, enter a name.
d) Expand **Nodes** to open the **Select Node** dialog box.
e) In the **Node ID** field, choose a node from the drop-down list.
f) In the **Router ID** field, enter the router ID.
g) Expand **Static Routes** to open the **Create Static Route** dialog box.
h) In the **Prefix** field, enter the IPv4 or IPv6 address.
i) Expand **Next Hop Addresses** and in the **Next Hop IP** field, enter the IPv4 or IPv6 address.
j) In the **Preference** field, enter a number, then click **UPDATE** and then **OK**.
k) In the **Select Node** dialog box, click **OK**.
l) In the **Create Node Profile** dialog box, click **OK**.
m) Check the **BGP**, **OSPF**, or **EIGRP** check boxes if desired, and click **NEXT**. Click **OK** to complete the Layer 3 configuration.

To confirm L3 configuration, in the **Navigation** pane, expand **Networking** > **VRFs**.

Deploying an Application Policy

Security Policy Enforcement

As traffic enters the leaf switch from the front panel interfaces, the packets are marked with the EPG of the source EPG. The leaf switch then performs a forwarding lookup on the packet destination IP address within the tenant space. A hit can result in any of the following scenarios:

1. A unicast (/32) hit provides the EPG of the destination endpoint and either the local interface or the remote leaf switch VTEP IP address where the destination endpoint is present.
2. A unicast hit of a subnet prefix (not /32) provides the EPG of the destination subnet prefix and either the local interface or the remote leaf switch VTEP IP address where the destination subnet prefix is present.
3. A multicast hit provides the local interfaces of local receivers and the outer destination IP address to use in the VXLAN encapsulation across the fabric and the EPG of the multicast group.

Note Multicast and external router subnets always result in a hit on the ingress leaf switch. Security policy enforcement occurs as soon as the destination EPG is known by the ingress leaf switch.

A miss result in the forwarding table causes the packet to be sent to the forwarding proxy in the spine switch. The forwarding proxy then performs a forwarding table lookup. If it is a miss, the packet is dropped. If it is a hit, the packet is sent to the egress leaf switch that contains the destination endpoint. Because the egress leaf switch knows the EPG of the destination, it performs the security policy enforcement. The egress leaf switch must also know the EPG of the packet source. The fabric header enables this process because it carries the EPG from the ingress leaf switch to the egress leaf switch. The spine switch preserves the original EPG in the packet when it performs the forwarding proxy function.

On the egress leaf switch, the source IP address, source VTEP, and source EPG information are stored in the local forwarding table through learning. Because most flows are bidirectional, a return packet populates the forwarding table on both sides of the flow, which enables the traffic to be ingress filtered in both directions.

Contracts Contain Security Policy Specifications

In the ACI security model, contracts contain the policies that govern the communication between EPGs. The contract specifies what can be communicated and the EPGs specify the source and destination of the communications. Contracts link EPGs, as shown below.

EPG 1 ---------- CONTRACT ---------- EPG 2

Endpoints in EPG 1 can communicate with endpoints in EPG 2 and vice versa if the contract allows it. This policy construct is very flexible. There can be many contracts between EPG 1 and EPG 2, there can be more than two EPGs that use a contract, and contracts can be reused across multiple sets of EPGs, and more.

There is also directionality in the relationship between EPGs and contracts. EPGs can either provide or consume a contract. An EPG that provides a contract is typically a set of endpoints that provide a service to a set of client devices. The protocols used by that service are defined in the contract. An EPG that consumes a contract is typically a set of endpoints that are clients of that service. When the client endpoint (consumer) tries to connect to a server endpoint (provider), the contract checks to see if that connection is allowed. Unless

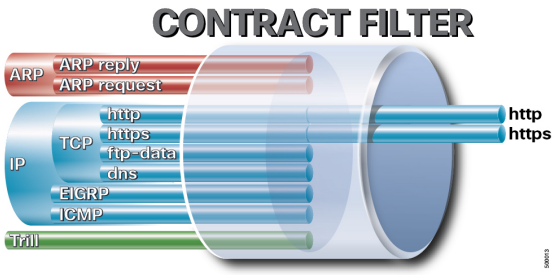
otherwise specified, that contract would not allow a server to initiate a connection to a client. However, another contract between the EPGs could easily allow a connection in that direction.

This providing/consuming relationship is typically shown graphically with arrows between the EPGs and the contract. Note the direction of the arrows shown below.

EPG 1 <-------consumes-------- CONTRACT <-------provides-------- EPG 2

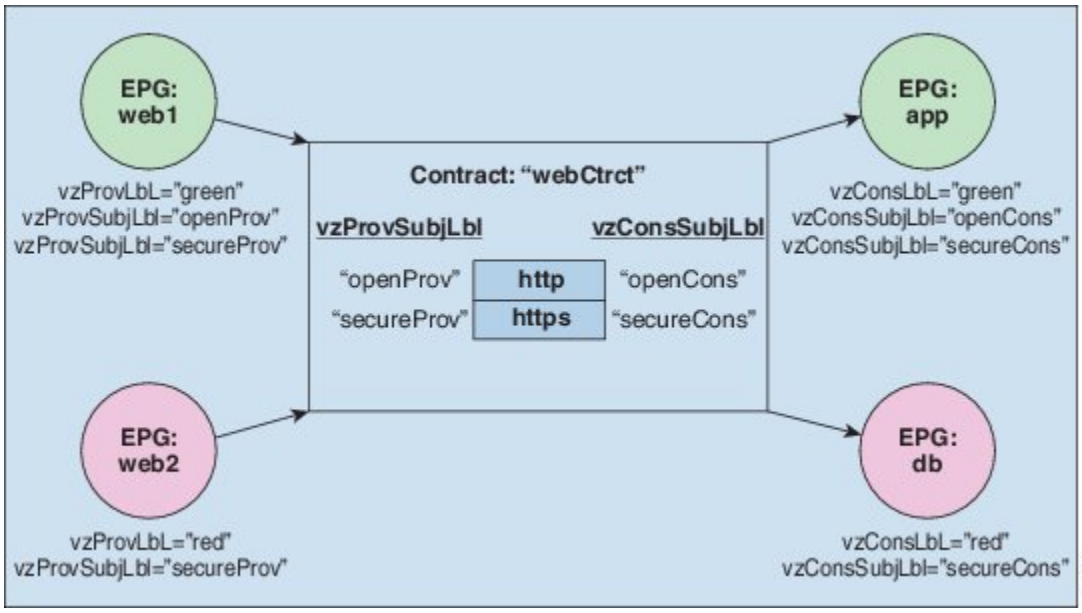
The contract is constructed in a hierarchical manner. It consists of one or more subjects, each subject contains one or more filters, and each filter can define one or more protocols.

Figure 22: Contract Filters



The following figure shows how contracts govern EPG communications.

Figure 23: Contracts Determine EPG to EPG Communications



For example, you may define a filter called HTTP that specifies TCP port 80 and port 8080 and another filter called HTTPS that specifies TCP port 443. You might then create a contract called webCtrct that has two sets of subjects. openProv and openCons are the subjects that contain the HTTP filter. secureProv and secureCons are the subjects that contain the HTTPS filter. This webCtrct contract can be used to allow both secure and non-secure web traffic between EPGs that provide the web service and EPGs that contain endpoints that want to consume that service.

These same constructs also apply for policies that govern virtual machine hypervisors. When an EPG is placed in a virtual machine manager (VMM) domain, the APIC downloads all of the policies that are associated with the EPG to the leaf switches with interfaces connecting to the VMM domain. For a full explanation of VMM domains, see the Virtual Machine Manager Domains chapter of the ACI Fundamentals manual. When this policy is created, the APIC pushes it (pre-populates it) to a VMM domain that specifies which switches allow connectivity for the endpoints in the EPGs. The VMM domain defines the set of switches and ports that allow endpoints in an EPG to connect to. When an endpoint comes on-line, it is associated with the appropriate EPGs. When it sends a packet, the source EPG and destination EPG are derived from the packet and the policy defined by the corresponding contract is checked to see if the packet is allowed. If yes, the packet is forwarded. If no, the packet is dropped.

The contract also allows more complex actions than just allow or deny. The contract can specify that traffic that matches a given subject can be re-directed to a service, can be copied, or can have its QoS level modified. With pre-population of the access policy in the concrete model, endpoints can move, new ones can come on-line, and communication can occur even if the APIC is off-line or otherwise inaccessible. The APIC is removed from being a single point of failure for the network. Upon packet ingress to the ACI fabric, security policies are enforced by the concrete model running in the switch.

Three-Tier Application Deployment

A filter specifies the data protocols to be allowed or denied by a contract that contains the filter. A contract can contain multiple subjects. A subject can be used to realize uni- or bidirectional filters. A unidirectional filter is a filter that is used in one direction, either from consumer-to-provider (IN) or from provider-to-consumer (OUT) filter. A bidirectional filter is the same filter that is used in both directions. It is not reflexive.

Contracts are policies that enable inter-End Point Group (inter-EPG) communication. These policies are the rules that specify communication between application tiers. If no contract is attached to the EPG, inter-EPG communication is disabled by default. No contract is required for intra-EPG communication because intra-EPG communication is always allowed.

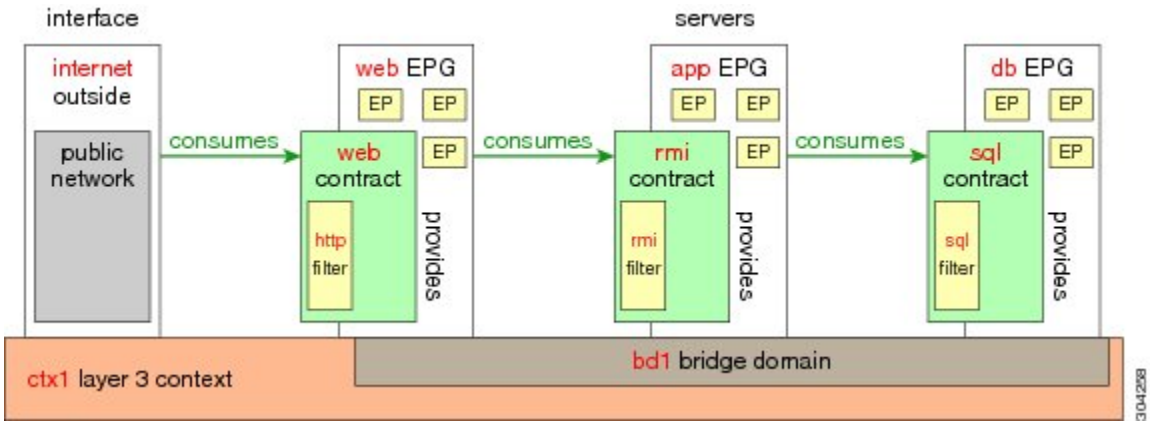
Application profiles enable you to model application requirements that the APIC then automatically renders in the network and data center infrastructure. The application profiles enable administrators to approach the resource pool in terms of applications rather than infrastructure building blocks. The application profile is a container that holds EPGs that are logically related to one another. EPGs can communicate with other EPGs in the same application profile and with EPGs in other application profiles.

To deploy an application policy, you must create the required application profiles, filters, and contracts. Typically, the APIC fabric hosts a three-tier application within a tenant network. In this example, the application is implemented by using three servers (a web server, an application server, and a database server). See the following figure for an example of a three-tier application.

The web server has the HTTP filter, the application server has the Remote Method Invocation (RMI) filter, and the database server has the Structured Query Language (SQL) filter. The application server consumes the SQL contract to communicate with the database server. The web server consumes the RMI contract to communicate with the application server. The traffic enters from the web server and communicates with the

application server. The application server then communicates with the database server, and the traffic can also communicate externally.

Figure 24: Three-Tier Application Diagram



Parameters to Create a Filter for http

The parameters to create a filter for http in this example is as follows:

Parameter Name	Filter for http
Name	http
Number of Entries	2
Entry Name	Dport-80 Dport-443
Ethertype	IP
Protocol	tcp tcp
Destination Port	http https

Parameters to Create Filters for rmi and sql

The parameters to create filters for rmi and sql in this example are as follows:

Parameter Name	Filter for rmi	Filter for sql
Name	rmi	sql

Parameter Name	Filter for rmi	Filter for sql
Number of Entries	1	1
Entry Name	Dport-1099	Dport-1521
Ethertype	IP	IP
Protocol	tcp	tcp
Destination Port	1099	1521

Example Application Profile Database

The application profile database in this example is as follows:

EPG	Provided Contracts	Consumed Contracts
web	web	rmi
app	rmi	sql
db	sql	--

Deploying an Application Policy Using the GUI

Creating a Filter Using the GUI

Create three separate filters. In this example they are HTTP, RMI, SQL. This task shows how to create the HTTP filter. The task is identical for creating the other filters.

Before You Begin

Verify that the tenant, network, and bridge domain have been created.

Procedure

Step 1 On the menu bar, choose **TENANTS**. In the **Navigation** pane, expand the **tenant** > **Security Policies**, right-click **Filters**, and click **Create Filter**.

Note In the **Navigation** pane, you expand the tenant where you want to add filters.

Step 2 In the **Create Filter** dialog box, perform the following actions:

a) In the **Name** field, enter the filter name (http).

b) Expand **Entries**, and in the **Name** field, enter the name (Dport-80).

c) From the **EtherType** drop-down list, choose the EtherType (IP).

d) From the **IP Protocol** drop-down list, choose the protocol (tcp).
e) From the **Destination Port/Range** drop-down lists, choose **http** in the **From** and **To** fields. (http)
f) Click **Update**, and click **Submit**.
The newly added filter appears in the **Navigation** pane and in the **Work** pane.

Step 3 Expand **Entries** in the **Name** field. Follow the same process to add another entry with HTTPS as the **Destination** port, and click **Update**.
This new filter rule is added.

Step 4 Follow the same process in the earlier steps to create two more filters (rmi and sql) and use the parameters provided in Parameters to Create Filters for rmi and sql, on page 139.

Creating a Contract Using the GUI

Procedure

Step 1 On the menu bar, choose **TENANTS** and the tenant name on which you want to operate. In the **Navigation** pane, expand the **tenant** > **Security Policies**.

Step 2 Right-click **Contracts** > **Create Contract**.

Step 3 In the **Create Contract** dialog box, perform the following tasks:
a) In the **Name** field, enter the contract name (web).
b) Click the + sign next to **Subjects** to add a new subject.
c) In the **Create Contract Subject** dialog box, enter a subject name in the **Name** field. (web)
d) **Note** This step associates the filters created that were earlier with the contract subject.
In the **Filter Chain** area, click the + sign next to **Filters**.
e) In the dialog box, from the drop-down menu, choose the filter name (http), and click **Update**.

Step 4 In the **Create Contract Subject** dialog box, click **OK**.

Step 5 Create two more contracts for rmi and for sql following the same steps in this procedure. For the rmi contract, choose the rmi subject and for sql, choose the sql subject.

Creating an Application Profile Using the GUI

Procedure

Step 1 On the menu bar, choose **TENANTS**. In the **Navigation** pane, expand the tenant, right-click **Application Profiles**, and click **Create Application Profile**.

Step 2 In the **Create Application Profile** dialog box, in the **Name** field, add the application profile name (OnlineStore).

Creating EPGs Using the GUI

The port the EPG uses must belong to one of the VM Managers (VMM) or physical domains associated with the EPG.

Procedure

Step 1 Expand **EPGs**. In the **Create Application EPG** dialog box, perform the following actions:

a) In the **Name** field, add the EPG name (db).
b) In the **Bridge Domain** field, choose the bridge domain from the drop-down list (bd1).
c) Check the **Associate to VM Domain Profiles** check box. Click **Next**.
d) In the **Step 2 for Specify the VM Domains** area, expand **Associate VM Domain Profiles** and from the drop-down list, choose the desired VMM domain. Click **Update** and click **OK**.

Step 2 In the **Create Application Profile** dialog box, create two more EPGs. The three EPGs should be db, app, and web in the same bridge domain and data center.

Consuming and Providing Contracts Using the GUI

You can associate contracts that were created earlier to create policy relationships between the EPGs.

When you name the provided and consumed contracts, verify that you give the same name for both provided and consumed contracts.

Procedure

Step 1 Click and drag across the APIC GUI window from the db EPG to the app EPG.
The **Add Consumed Contract** dialog box is displayed.

Note The db, app, and web EPGs are displayed as icons.

Step 2 In the **Name** field, from the drop-down list, choose **sql** contract. Click **OK**.
This step enables the db EPG to provide the sql contract and the app EPG to consume the sql contract.

Step 3 Click and drag across the APIC GUI screen from the app ePG to the web EPG.
The **Add Consumed Contract** dialog box is displayed.

Step 4 In the **Name** field, from the drop-down list, choose **rmi** contract. Click **OK**.
This step enables the app EPG to provide the rmi contract and the web EPG to consume the rmi contract.

Step 5 Click the web EPG icon, and click the + sign in the **Provided Contracts** area.
The **Add Provided Contract** dialog box is displayed.

Step 6 In the **Name** field, from the drop-down list, choose **web** contract. Click **OK**. Click **Submit**.
You have created a three-tier application profile called OnlineStore.

Step 7 To verify, in the **Navigation** pane, navigate to and click **OnlineStore** under **Application Profiles**.
In the **Work** pane, you can see the three EPGs app, db, and web are displayed.

Step 8 In the **Work** pane, choose **Operational** > **Contracts**.
You can see the EPGs and contracts displayed in the order that they are consumed and provided.

Statically Deploying an EPG on a Specific Port

This topic provides a typical example of how to statically deploy an EPG on a specific port when using Cisco APIC.

Deploying an EPG on a Specific Port with APIC Using the GUI

Before You Begin

The tenant where you deploy the EPG is already created.

Procedure

Step 1 On the menubar, click **TENANTS**.

Step 2 In the **Navigation** pane, expand the appropriate ***Tenant_name*** > **Application Profiles**.

Step 3 Right-click **Application Profiles** and click **Create Application Profile**.

Step 4 In the **Create Application Profile** dialog box, perform the following actions:

a) In the **Name** field, enter a name for the application profile.

b) Expand **EPGs**.

c) In the **Create Application EPG** dialog box, in the **Name** field, enter an **EPG** name.

d) In the **Statically Link with Leaves/Paths** field, check the checkbox for **Statically Link with Leaves/Paths**. (this is selected to specify on which port the EPG is required to be deployed). Click Next.

e) In the **Leaves/Paths** area, expand **Paths**.
In this example we are deploying the EPG on the port of a node. Alternatively, you could choose to deploy the EPG on a node.

f) From the **Path** drop-down list, choose the appropriate node and port.

g) In the **Encap** field, enter the VLAN to be deployed.

h) In the **Deployment Immediacy** field drop-down list, choose the preferred deployment time.

i) In the **Mode** field, choose the appropriate mode.

j) Click **OK**, and click **Submit**.

Step 5 In the **Navigation** pane, expand **Application Profiles** to view the new application profile.

Step 6 Expand **Application EPGs**, to view the new EPG.

Step 7 Expand the EPG and click **Static Bindings (Paths)**, and in the **Work** pane, view the details of the static binding paths that are established.

Deploying an EPG on a Specific Port with APIC Using the NX-OS Style CLI

Procedure

Step 1 Configure a VLAN domain:

Example:

```
apic1(config)# vlan-domain dom1
apic1(config-vlan)# vlan 10-100
```

Step 2 Create a tenant:

Example:

```
apic1# configure
apic1(config)# tenant t1
```

Step 3 Create a private network/VRF:

Example:

```
apic1(config-tenant)# vrf context ctx1
apic1(config-tenant-vrf)# exit
```

Step 4 Create a bridge domain:

Example:

```
apic1(config-tenant)# bridge-domain bd1
apic1(config-tenant-bd)# vrf member ctx1
apic1(config-tenant-bd)# exit
```

Step 5 Create an application profile and an application EPG:

Example:

```
apic1(config-tenant)# application AP1
apic1(config-tenant-app)# epg EPG1
apic1(config-tenant-app-epg)# bridge-domain member bd1
apic1(config-tenant-app-epg)# exit
apic1(config-tenant-app)# exit
apic1(config-tenant)# exit
```

Step 6 Associate the EPG with a specific port:

Example:

```
apic1(config)# leaf 1017
apic1(config-leaf)# interface ethernet 1/13
apic1(config-leaf-if)# vlan-domain member dom1
apic1(config-leaf-if)# switchport trunk allowed vlan 20 tenant t1 application AP1 epg EPG1
```

Note The vlan-domain and vlan-domain member commands mentioned in the above example are a pre-requisite for deploying an EPG on a port.

Deploying an EPG on a Specific Port with APIC Using the REST API

Before You Begin

The tenant where you deploy the EPG is created.

Procedure

Deploy an EPG on a specific port.

Example:

```
<fvTenant name="<tenant_name>" dn="uni/tn-test1" >
    <fvCtx name="<network_name>" pcEnfPref="enforced" knwMcastAct="permit"/>
    <fvBD name="<bridge_domain_name>" unkMcastAct="flood" >
        <fvRsCtx tnFvCtxName="<network_name>"/>
    </fvBD>
    <fvAp name="<application_profile>" >
        <fvAEPg name="<epg_name>" >
            <fvRsPathAtt tDn="topology/pod-1/paths-1017/pathep-[eth1/13]" mode="regular"
instrImedcy="immediate" encap="vlan-20"/>
        </fvAEPg>
    </fvAp>
</fvTenant>
```

Creating Domains, Attach Entity Profiles, and VLANs to Deploy an EPG on a Specific Port

This topic provides a typical example of how to create physical domains, Attach Entity Profiles (AEP), and VLANs that are mandatory to deploy an EPG on a specific port.

Note All endpoint groups (EPGs) require a domain. Interface policy groups must also be associated with Attach Entity Profile (AEP), and the AEP must be associated with a domain, if the AEP and EPG have to be in same domain. Based on the association of EPGs to domains and of interface policy groups to domains, the ports and VLANs that the EPG uses are validated. The following domain types associate with EPGs:

- Application EPGs
- Layer 3 external outside network instance EPGs
- Layer 2 external outside network instance EPGs
- Management EPGs for out-of-band and in-band access

The APIC checks if an EPG is associated with one or more of these types of domains. If the EPG is not associated, the system accepts the configuration but raises a fault. The deployed configuration may not function properly if the domain association is not valid. For example, if the VLAN encapsulation is not valid for use with the EPG, the deployed configuration may not function properly.

Creating Domains, and VLANS to Deploy an EPG on a Specific Port Using the GUI

Before You Begin

- The tenant where you deploy the EPG is already created.
- An EPG is statically deployed on a specific port.

Procedure

Step 1 On the menu bar, click **FABRIC** > **Access Policies**.

Step 2 In the **Navigation** pane, click **Quick Start**.

Step 3 In the **Work** pane, click **Configure an interface, PC, and vPC**.

Step 4 In the **Configure Interface, PC, and vPC** dialog box, click the + icon to select switches and perform the following actions:

a) From the **Switches** drop-down list, check the check box for the desired switch.

b) In the **Switch Profile Name** field, a switch name is automatically populated.

Note Optionally, you can enter a modified name.

c) Click the + icon to configure the switch interfaces.

d) In the **Interface Type** field, click the **Individual** radio button.

e) In the **Interfaces** field, enter the range of desired interfaces.

f) In the **Interface Selector Name** field, an interface name is automatically populated.

Note Optionally, you can enter a modified name.

g) In the **Interface Policy Group** field, choose the **Create One** radio button.

h) From the **Link Level Policy** drop-down list, choose the appropriate link level policy.

Note Create additional policies as desired, otherwise the default policy settings are available.

i) From the **Attached Device Type** field, choose the appropriate device type.

j) In the **Domain** field, click the **Create One** radio button.

k) In the **Domain Name** field, enter a domain name.

l) In the **VLAN** field, click the **Create One** radio button.

m) In the **VLAN Range** field, enter the desired VLAN range. Click **Save**, and click **Save** again.

n) Click **Submit**.

Step 5 On the menu bar, click **TENANTS**. In the **Navigation** pane, expand the appropriate ***Tenant_name*** > **Application Profiles** > **Domains (VMs and Bare-Metals)** > ***EPG_name*** and perform the following actions:

a) Right-click **Domains (VMs and Bare-Metals)**, and click **Add Physical Domain Association**.

b) In the **Add Physical Domain Association** dialog box, from the **Physical Domain Profile** drop-down list, choose the appropriate domain.

c) In the **Deploy Immediacy** field, click the desired radio button.

d) In the **Resolution Immediacy** field, click the desired radio button. Click **Submit**.
The AEP is associated with a specific port on a node and with a domain. The physical domain is associated with the VLAN pool and the Tenant is associated with this physical domain.

The switch profile and the interface profile are created. The policy group is created in the port block under the interface profile. The AEP is automatically created, and it is associated with the port block and with the domain. The domain is associated with the VLAN pool and the Tenant is associated with the domain.

Creating AEP, Domains, and VLANs to Deploy an EPG on a Specific Port Using the NX-OS Style CLI

Before You Begin

- The tenant where you deploy the EPG is already created.
- An EPG is statically deployed on a specific port.

Procedure

Step 1 Create a VLAN domain and assign VLAN ranges:

Example:

```
apic1(config)# vlan-domain domP
apic1(config-vlan)# vlan 10
apic1(config-vlan)# vlan 25
apic1(config-vlan)# vlan 50-60
apic1(config-vlan)# exit
```

Step 2 Create an interface policy group and assign a VLAN domain to the policy group:

Example:

```
apic1(config)# template policy-group PortGroup
apic1(config-pol-grp-if)# vlan-domain member domP
```

Step 3 Create a leaf interface profile, assign an interface policy group to the profile, and assign the interface IDs on which the profile will be applied:

Example:

```
apic1(config)# leaf-interface-profile InterfaceProfile1
apic1(config-leaf-if-profile)# leaf-interface-group range
apic1(config-leaf-if-group)# policy-group PortGroup
apic1(config-leaf-if-group)# interface ethernet 1/11-13
apic1(config-leaf-if-profile)# exit
```

Step 4 Create a leaf profile, assign the leaf interface profile to the leaf profile, and assign the leaf IDs on which the profile will be applied:

Example:

```
apic1(config)# leaf-profile SwitchProfile-1019
apic1(config-leaf-profile)# leaf-interface-profile InterfaceProfile1
apic1(config-leaf-profile)# leaf-group range
```

```
apic1(config-leaf-group)# leaf 1019
apic1(config-leaf-group)#
```

Creating AEP, Domains, and VLANs to Deploy an EPG on a Specific Port Using the REST API

Before You Begin

- The tenant where you deploy the EPG is already created.
- An EPG is statically deployed on a specific port.

Procedure

Step 1 Create the interface profile, switch profile and the Attach Entity Profile (AEP).

Example:

```
   <infraInfra>

      <infraNodeP  name="<switch_profile_name>" dn="uni/infra/nprof-<switch_profile_name>"
 >
           <infraLeafS  name="SwitchSeletor" descr="" type="range">
                <infraNodeBlk name="nodeBlk1" descr="" to_="1019" from_="1019"/>
           </infraLeafS>
          <infraRsAccPortP tDn="uni/infra/accportprof-<interface_profile_name>"/>
      </infraNodeP>

      <infraAccPortP name="<interface_profile_name>"
dn="uni/infra/accportprof-<interface_profile_name>" >
           <infraHPortS  name="portSelector"  type="range">
                <infraRsAccBaseGrp tDn="uni/infra/funcprof/accportgrp-<port_group_name>"
fexId="101"/>
              <infraPortBlk name="block2"  toPort="13" toCard="1" fromPort="11"
fromCard="1"/>
         </infraHPortS>
     </infraAccPortP>

    <infraAccPortGrp  name="<port_group_name>"
dn="uni/infra/funcprof/accportgrp-<port_group_name>" >
          <infraRsAttEntP tDn="uni/infra/attentp-<attach_entity_profile_name>"/>
          <infraRsHIfPol tnFabricHIfPolName="1GHifPol"/>
    </infraAccPortGrp>

    <infraAttEntityP  name="<attach_entity_profile_name>"
dn="uni/infra/attentp-<attach_entity_profile_name>" >
         <infraRsDomP tDn="uni/phys-<physical_domain_name>"/>
    </infraAttEntityP>

<infraInfra>
```

Step 2 Create a domain.

Example:

```
<physDomP  name="<physical_domain_name>" dn="uni/phys-<physical_domain_name>">
    <infraRsVlanNs tDn="uni/infra/vlanns-[<vlan_pool_name>]-static"/>
</physDomP>
```

Step 3 Create a VLAN range.

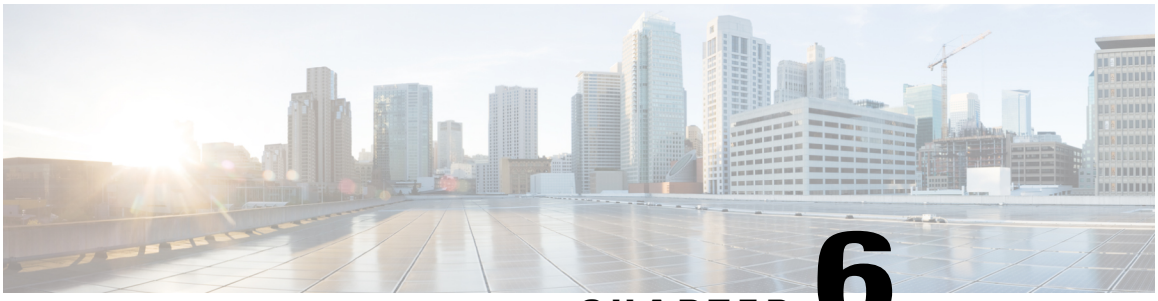
Example:

```
<fvnsVlanInstP  name="<vlan_pool_name>" dn="uni/infra/vlanns-[<vlan_pool_name>]-static"
allocMode="static">
    <fvnsEncapBlk name="" descr="" to="vlan-25" from="vlan-10"/>
</fvnsVlanInstP>
```

Step 4 Associate the EPG with the domain.

Example:

```
<fvTenant  name="<tenant_name>" dn="uni/tn-" >
    <fvAEPg prio="unspecified" name="<epg_name>" matchT="AtleastOne"
dn="uni/tn-test1/ap-AP1/epg-<epg_name>" descr="">
        <fvRsDomAtt tDn="uni/phys-<physical_domain_name>" instrImedcy="immediate"
resImedcy="immediate"/>
    </fvAEPg>
</fvTenant>
```



CHAPTER 6

ACI Fabric Layer 3 Outside Connectivity

This chapter contains the following sections:

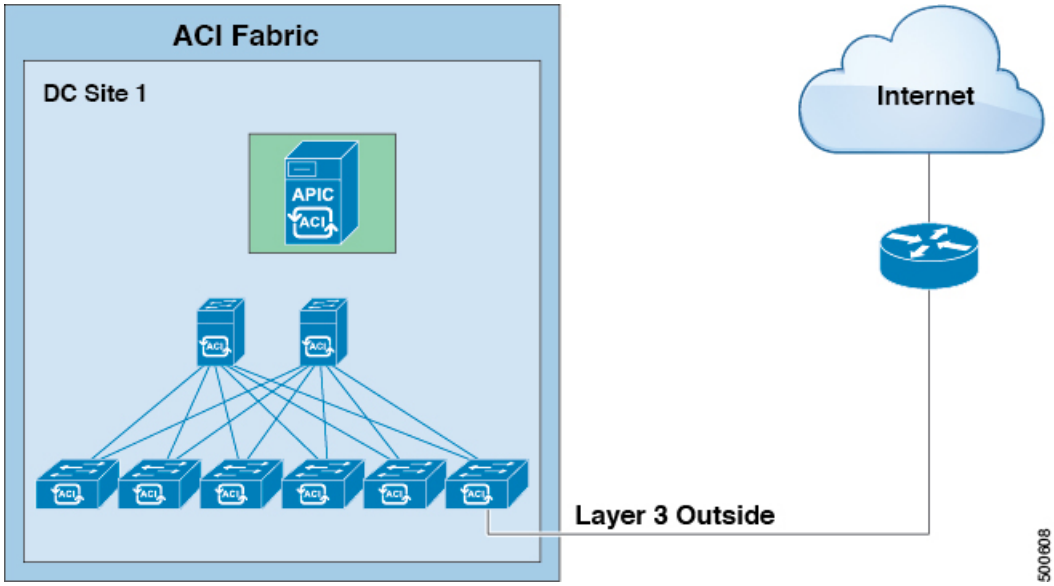
Layer 3 Workflows

-

ACI Layer 3 Outside Network Workflows

This workflow provides an overview of the steps required to configure a Layer 3 outside network connection.

Figure 25: Layer 3 outside network connection



1. Prerequisites

- Ensure that you have read/write access privileges to the infra security domain.
- Ensure that the target leaf switches with the necessary interfaces are available.

Configure a Layer 3 Outside Network

Choose which of these management access scenarios you will use:

- For a Layer 3 Outside that will be consumed within a single tenant, follow the instructions for configuring BGP or OSPF.
- For a Layer 3 Outside that will be consumed (shared) among multiple tenants, follow "Layer 3 Outside guidelines.
- For Layer 3 Outside transit routing uses cases, follow ACI transit routing instructions.

Recommended topics

For additional information, see the following topics:

- Guidelines for Configuring a BGP Layer 3 Outside Network Connection, on page 153

• Shared Layer 3 Out, on page 166
• ACI Transit Routing, on page 177

Guidelines for Configuring a BGP Layer 3 Outside Network Connection

When configuring a BGP external routed network, follow these guidelines:

- Whenever a router ID is created on a leaf switch, it creates an internal loopback address. When setting up a BGP connection on a leaf switch, your router ID cannot be the same as the interface IP address as it is not supported on the ACI leaf switch. The router ID must be a different address in a different subnet. On the external Layer 3 device, the router ID can be the loopback address or an interface address. Ensure that the route to leaf router ID is present in the routing table of the the Layer3 device either through static route or OSPF configuration. Also, when setting up the BGP neighbor on a Layer 3 device, the peer IP address that is used must be the router ID of the leaf switch.
- While configuring two external Layer 3 networks with BGP on the same node, loopback addresses must be explicitly defined. Failing to follow this guideline can prevent BGP from being established.
- By definition, the router ID is a loopback interface. To change the router ID and assign a different address for loopback, you must create a loopback interface policy. (The loopback policy can be configured as one for each address family, IPv4 and IPv6.) If you do not wish to create a loopback policy, then you can enable a router ID loopback which is enabled by default. If the router ID loopback is disabled, no loopback is created for the specific Layer 3 outside on which it is deployed.
- This configuration task is applicable for iBGP and eBGP. If the BGP configuration is on a loopback address then it can be an iBGP session or a multi-hop eBGP session. If the peer IP address is for a physical interface where the BGP peer is defined, then the physical interface is used.
- The user must configure an IPv6 address to enable peering over loopback using IPv6.
- The autonomous system feature can only be used for eBGP peers. It enables a router to appear to be a member of a second autonomous system (AS), in addition to its real AS. Local AS allows two ISPs to merge without modifying peering arrangements. Routers in the merged ISP become members of the new autonomous system but continue to use their old AS numbers for their customers.
- Starting with release 1.2(1x), tenant networking protocol policies for BGP `l3extOut` connections can be configured with a maximum prefix limit that enables monitoring and restricting the number of route prefixes received from a peer. Once the max prefix limit is exceeded, a log entry can be recorded, further prefixes can be rejected, the connection can be restarted if the count drops below the threshold in a fixed interval, or the connection is shut down. Only one option can be used at a time. The default setting is a limit of 20,000 prefixes, after which new prefixes are rejected. When the reject option is deployed, BGP accepts one more prefix beyond the configured limit and the APIC raises a fault.

Note When you configure Layer 3 Outside (L3Out) connections to external routers, it is critical that the MTU be set appropriately on both sides. On some platforms, such as ACI, Cisco NX-OS, and Cisco IOS, the configurable MTU value takes into account packet headers (resulting in a max packet size to be set as 9000 bytes), whereas other platforms such as IOS-XR configure the MTU value exclusive of packet headers (resulting in a max packet size of 8986 bytes). For the appropriate MTU values for each platform, see the relevant configuration guides. Cisco highly recommends you test the MTU using CLI-based commands. For example, on the Cisco NX-OS CLI, use a command such as `ping 1.1.1.1 df-bit packet-size 9000 source-interface ethernet 1/1`

BGP Connection Types and Loopback Guidelines

For BGP connection types and loopback set up requirements, follow these guidelines:

- When a router ID is created for a node, a loopback interface with the same IP address as the router ID is also created. This is the default behavior but can be overridden when configuring the router ID.
- The IP address configured for the router ID should be a different address in a different subnet from any other IP address configured on the node.
- The loopback interface with the router ID IP address can be used for peering with an external router if there is only one external BGP peer per node. When peering with multiple BGP peers on the same node, the router ID loopback address must not be used. An explicit loopback interface policy per BGP peer must be used.
- A loopback interface policy is not required when peering with an external router on a directly connected network.
- When peering to an external router with a loopback interface (iBGP or eBGP multi-hop) a static route or OSPF route is required to reach the remote peer loopback address.
- For BGP, the loopback creation is selected by default. When it is selected, the loopack is used as the source interface to establish BGP sessions. However, to establish eBGP over a physical interface, the administrator must not create loopback.

Table 2:

BGP Connection Type	Loopback required	Loopback same as Router ID	Static/OSPF route required
iBGP direct	No	Not applicable	No
iBGP loopback peering	Yes, a separate loopback per BGP peer	No, if multiple Layer 3 out are on the same node	Yes
eBGP direct	No	Not applicable	No

BGP Connection Type	Loopback required	Loopback same as Router ID	Static/OSPF route required
eBGP loopback peering (multi-hop)	Yes, a separate loopback per BGP peer	No, if multiple Layer 3 out are on the same node	Yes

Configuring BGP External Routed Network Using the GUI

Before You Begin

The tenant, VRF, and bridge domain where you configure the BGP external routed network is already created.

Procedure

Step 1 In the **Navigation** pane, expand **Tenant_name** > **Networking** > **External Routed Networks**.

Step 2 Right-click, and click **Create Routed Outside**.

Step 3 In the **Create Routed Outside** dialog box, perform the following actions:

a) In the **Name** field, enter a name for the external routed network policy.

b) Click the **BGP** checkbox.

Note BGP peer reachability must be available in one of two ways. You must either configure static routes or enable OSPF.

c) (Optional) In the **Route Control Enforcement** field, check the **Import** check box.

Note Check this check box if you wish to enforce import control with BGP.

d) From the **VRF** field drop-down list, choose the desired VRF.

e) Expand the **Route Control for Dampening** field, and choose the desired address family type and route dampening policy. Click **Update**.
In this step, the policy can be created either with step 4 or there is also an option to **Create route profile** in the drop-down list where the policy name is selected.

f) Expand **Nodes and Interfaces Protocol Policies**.

g) In the **Create Node Profile** dialog box, enter a name for the node profile.

h) Expand **Nodes**.

i) From the **Select Node** dialog box, from the **Node ID** field drop-down list, choose a node.

j) In the **Router ID** field, enter the router ID.

k) Expand **Loopback Address**, and in the **IP** field, enter the IP address. Click **Update**.

Note Enter an IPv6 address. If you did not add the router ID in the earlier step, you can add an IPv4 address in the **IP** field.

l) Click **OK**.

Step 4 In the Navigation pane, expand **Tenant_name** > **Networking** > **Route Profiles**. Right-click **Route Profiles**, and click **Create Route Profile**. In the **Create Route Profile** dialog box, perform the following actions:

a) In the **Name** field, enter a name for the route control VRF.

b) Expand the **Create Route Control Context** dialog box.

c) In the **Name** field, enter a name for the route control VRF.

d) From the **Set Attribute** drop-down list, choose **Create Action Rule Profile**.
When creating an action rule, set the route dampening attributes as desired.

Step 5 In the **Create Interface Profiles** dialog box, perform the following actions:

a) In the **Name** field, enter an interface profile name.
b) In the **Interfaces** area, choose the desired interface tab, and then expand the interface.

Step 6 In the **Select Routed Interface** dialog box, perform the following actions:

a) From the **Path** field drop-down list, choose the node and the interface.
b) In the **IP Address** field, enter the IP address.
Note Depending upon your requirements, you can add an IPv6 address or an IPv4 address.
c) (Optional) If you entered an IPv6 address in the earlier step, in the **Link-local Address** field, enter an IPv6 address.
d) Expand **BGP Peer Connectivity Profile** field.

Step 7 In the **Create Peer Connectivity Profile** dialog box, perform the following actions:

a) In the **Peer Address** field, the dynamic neighbor feature is available. If desired by the user, any peer within a specified subnet can communicate or exchange routes with BGP.
Enter an IPv4 of an IPv6 address to correspond with IPv4 or IPv6 addresses entered in the earlier in the steps.
b) In the **BGP Controls** field, check the desired controls.
c) In the **Autonomous System Number** field, choose the desired value.
d) (Optional) In the **Weight for routes from this neighbor** field, choose the desired value.
e) (Optional) In the **Private AS Control** field, check the check box for **Remove AS**.
f) (Optional) In the **Local Autonomous System Number Config** field, choose the desired value.
Optionally required for the local autonomous system feature for eBGP peers.
g) (Optional) In the **Local Autonomous System Number** field, choose the desired value.
Optionally required for the local autonomous system feature for eBGP peers.
Note The value in this field must not be the same as the value in the **Autonomous System Number** field.
h) Click **OK**.

Step 8 Perform the following actions:

a) In the **Select Routed Interface** dialog box, click **OK**.
b) In the **Create Interface Profile** dialog box, click **OK**.
c) In the **Create Node Profile** dialog box, click **OK**.
The **External EPG Networks** area is displayed.
d) In **Create Routed Outside** dialog box, choose the node profile you created earlier, and click **Next**.

Step 9 Expand **External EPG Networks**, and in the **Create External Network** dialog box, perform the following actions:

a) In the **Name** field, enter a name for the external network.
b) Expand **Subnet**.
c) In the **Create Subnet** dialog box, in the **IP address** field, enter the subnet addresses as required.
Note Enter an IPv4 or IPv6 address depending upon what you have entered in earlier steps.
When creating the external subnet, you must configure either both the BGP loopbacks in the prefix EPG or neither of them. If you configure only one BGP loopback, then BGP neighborship is not established.

d) In the **Scope** field, check the check boxes for **Export Route Control Subnet**, **Import Route Control Subnet**, and **Security Import Subnet**. Click **OK**.

Note Check the **Import Route Control Subnet** check box if you wish to enforce import control with BGP.

Step 10 In the **Create External Network** dialog box, click **OK**.

Step 11 In the **Create Routed Outside** dialog box, click **Finish**.
The eBGP is configured for external connectivity.

Configuring BGP External Routed Network Using the REST API

Before You Begin

The tenant where you configure the BGP external routed network is already created.

Procedure

The following shows how to configure the BGP external routed network using the REST API:

Example:

```
<l3extOut descr="" dn="uni/tn-t1/out-l3out-bgp" enforceRtctrl="export" name="l3out-bgp"
ownerKey="" ownerTag="" targetDscp="unspecified">
<l3extRsEctx tnFvCtxName="ctx3"/>
<l3extLNodeP configIssues="" descr="" name="l3extLNodeP_1" ownerKey="" ownerTag=""
tag="yellow-green" targetDscp="unspecified">
<l3extRsNodeL3OutAtt rtrId="1.1.1.1" rtrIdLoopBack="no" tDn="topology/pod-1/node-101"/>
<l3extLIfP descr="" name="l3extLIfP_2" ownerKey="" ownerTag="" tag="yellow-green">
<l3extRsNdIfPol tnNdIfPolName=""/>
<l3extRsIngressQosDppPol tnQosDppPolName=""/>
<l3extRsEgressQosDppPol tnQosDppPolName=""/>
<l3extRsPathL3OutAtt addr="3001::31:0:1:2/120" descr="" encap="vlan-3001" encapScope="local"
 ifInstT="sub-interface" llAddr="::" mac="00:22:BD:F8:19:FF" mode="regular" mtu="inherit"
tDn="topology/pod-1/paths-101/pathep-[eth1/8]" targetDscp="unspecified">
<bgpPeerP addr="3001::31:0:1:0/120" allowedSelfAsCnt="3" ctrl="send-com,send-ext-com"
descr="" name="" peerCtrl="bfd" privateASctrl="remove-all,remove-exclusive,replace-as"
ttl="1" weight="1000">
<bgpRsPeerPfxPol tnBgpPeerPfxPolName=""/>
<bgpAsP asn="3001" descr="" name=""/>
</bgpPeerP>
</l3extRsPathL3OutAtt>
</l3extLIfP>
<l3extLIfP descr="" name="l3extLIfP_1" ownerKey="" ownerTag="" tag="yellow-green">
<l3extRsNdIfPol tnNdIfPolName=""/>
<l3extRsIngressQosDppPol tnQosDppPolName=""/>
<l3extRsEgressQosDppPol tnQosDppPolName=""/>
<l3extRsPathL3OutAtt addr="31.0.1.2/24" descr="" encap="vlan-3001" encapScope="local"
ifInstT="sub-interface" llAddr="::" mac="00:22:BD:F8:19:FF" mode="regular" mtu="inherit"
tDn="topology/pod-1/paths-101/pathep-[eth1/8]" targetDscp="unspecified">
<bgpPeerP addr="31.0.1.0/24" allowedSelfAsCnt="3" ctrl="send-com,send-ext-com" descr=""
name="" peerCtrl="" privateASctrl="remove-all,remove-exclusive,replace-as" ttl="1"
weight="100">
<bgpRsPeerPfxPol tnBgpPeerPfxPolName=""/>
<bgpLocalAsnP asnPropagate="none" descr="" localAsn="200" name=""/>
<bgpAsP asn="3001" descr="" name=""/>
</bgpPeerP>
</l3extRsPathL3OutAtt>
</l3extLIfP>
</l3extLNodeP>
```

```
<l3extRsL3DomAtt tDn="uni/l3dom-l3-dom"/>
<l3extRsDampeningPol af="ipv6-ucast" tnRtctrlProfileName="damp_rp"/>
<l3extRsDampeningPol af="ipv4-ucast" tnRtctrlProfileName="damp_rp"/>
<l3extInstP descr="" matchT="AtleastOne" name="l3extInstP_1" prio="unspecified"
targetDscp="unspecified">
<l3extSubnet aggregate="" descr="" ip="130.130.130.0/24" name="" scope="import-rtctrl">
</l3extSubnet>
<l3extSubnet aggregate="" descr="" ip="130.130.131.0/24" name="" scope="import-rtctrl"/>
<l3extSubnet aggregate="" descr="" ip="120.120.120.120/32" name=""
scope="export-rtctrl,import-security"/>
<l3extSubnet aggregate="" descr="" ip="3001::130:130:130:100/120" name=""
scope="import-rtctrl"/>
</l3extInstP>
<bgpExtP descr=""/>
</l3extOut>
<rtctrlProfile descr="" dn="uni/tn-t1/prof-damp_rp" name="damp_rp" ownerKey="" ownerTag=""
 type="combinable">
 <rtctrlCtxP descr="" name="ipv4_rpc" order="0">
  <rtctrlScope descr="" name="">
   <rtctrlRsScopeToAttrP tnRtctrlAttrPName="act_rule"/>
  </rtctrlScope>
 </rtctrlCtxP>
</rtctrlProfile>
<rtctrlAttrP descr="" dn="uni/tn-t1/attr-act_rule" name="act_rule">
  <rtctrlSetDamp descr="" halfLife="15" maxSuppressTime="60" name="" reuse="750"
suppress="2000" type="dampening-pol"/>
</rtctrlAttrP>
```

Configuring BGP External Routed Network Using the NX-OS Style CLI

Procedure

The following shows how to configure the BGP external routed network using the NX-OS CLI:

Example:

```
apic1(config-leaf)#template route-profile damp_rp tenant t1
This template will be available on all leaves where tenant t1 has a VRF deployment
apic1(config-leaf-template-route-profile)#set dampening 15 750 2000 60
apic1(config-leaf-template-route-profile)#exit
apic1(config-leaf)#
apic1(config-leaf)#router bgp 100
apic1(config-bgp)#vrf member tenant t1 vrf ctx3
apic1(config-leaf-bgp-vrf)# neighbor 32.0.1.0/24 l3out l3out-bgp
apic1(config-leaf-bgp-vrf-neighbor)#update-source ethernet 1/16.401
apic1(config-leaf-bgp-vrf-neighbor)#address-family ipv4 unicast
apic1(config-leaf-bgp-vrf-neighbor-af)#weight 400
apic1(config-leaf-bgp-vrf-neighbor-af)#exit
apic1(config-leaf-bgp-vrf-neighbor)#remote-as 65001
apic1(config-leaf-bgp-vrf-neighbor)#private-as-control remove-exclusive
apic1(config-leaf-bgp-vrf-neighbor)#private-as-control remove-exclusive-all
apic1(config-leaf-bgp-vrf-neighbor)#private-as-control remove-exclusive-all-replace-as
apic1(config-leaf-bgp-vrf-neighbor)#exit
apic1(config-leaf-bgp-vrf)# address-family ipv4 unicast
apic1(config-leaf-bgp-vrf-af)#inherit bgp dampening damp_rp
This template will be inherited on all leaves where VRF ctx3 has been deployed
apic1(config-leaf-bgp-vrf-af)#exit
apic1(config-leaf-bgp-vrf)# address-family ipv6 unicast
apic1(config-leaf-bgp-vrf-af)#inherit bgp dampening damp_rp
This template will be inherited on all leaves where VRF ctx3 has been deployed
apic1(config-leaf-bgp-vrf-af)#exit
```

Configuring a Tenant Layer 3 Outside Network Connection

This topic provides a typical example of how to configure a Layer 3 Outside for tenant networks when using Cisco APIC.

Note When you configure Layer 3 Outside (L3Out) connections to external routers, it is critical that the MTU be set appropriately on both sides. On some platforms, such as ACI, Cisco NX-OS, and Cisco IOS, the configurable MTU value takes into account packet headers (resulting in a max packet size to be set as 9000 bytes), whereas other platforms such as IOS-XR configure the MTU value exclusive of packet headers (resulting in a max packet size of 8986 bytes). For the appropriate MTU values for each platform, see the relevant configuration guides. Cisco highly recommends you test the MTU using CLI-based commands. For example, on the Cisco NX-OS CLI, use a command such as `ping 1.1.1.1 df-bit packet-size 9000 source-interface ethernet 1/1`

Configuring a Layer 3 Outside for Tenant Networks Using the GUI

The external routed network configured in the example can also be extended to support IPv4. Both IPv4 and IPv6 routes can be advertised to and learned from the external routed network.

Before You Begin

- The tenant, VRF, and bridge domain are created.
- The external routed domain is created.

Procedure

Step 1 On the menubar, click **TENANTS**.

Step 2 In the **Navigation** pane, expand the ***Tenant_name*** > **Networking** > **External Routed Networks** and perform the following actions:

a) Right-click **External Routed Networks** and click **Create Routed Outside**.
b) In the **Create Routed Outside** dialog box, in the **Name** field, enter a name for the routed outside.
c) In the area with the routing protocol check boxes, check the desired protocol.
The options available are BGP, OSPF, EIGRP. Later in the steps, this will make available, the route summarization policy in the **Create External Network** dialog box.
d) In the **VRF** field, from the drop-down list, choose the appropriate VRF.
e) From the **External Routed Domain** drop-down list, choose the appropriate external routed domain.
f) Check the checkbox for the desired protocol.
Depending on the protocol you choose, the properties must be set.
g) Expand **Nodes and Interfaces Protocol Profile**.
h) In the **Create Node Profile** dialog box, in the **Name** field, enter a name.
i) Expand **Nodes**.
j) In the **Select Node** dialog box, from the **Node ID** drop-down menu, choose the appropriate node ID.

k) In the **Router ID** field, enter the router ID.
l) Expand **Loopback Addresses**, in the IP field, enter the IP address. Click **Update**.
Note In the **Loopback Addresses** fields, create an IPv4 and/or IPv6 loopback as desired.
m) Click **OK**.

Step 3 Expand **Interface Profiles**, and perform the following actions:
a) In the **Create Interface Profile** dialog box, in the **Name** field, enter a name for the profile.
b) Expand **Routed Interfaces**.
c) In the **Select Routed Interface** dialog box, from the **Path** drop-down list, choose the interface path.
d) In the **IP Address** field, enter the IP address.
Note To configure IPv6, you must enter the link-local address in the **Link-local Address** field in the dialog box.
e) Click **OK**.
The routed interface details are displayed in the **Create Interface Profile** dialog box.
f) Click **OK**.

Step 4 In the **Create Node Profile** dialog box, click **OK**.

Step 5 In the **Create Routed Outside** dialog box, click **Next**.

Step 6 In the **External EPG Networks** area, expand **External EPG Networks**.

Step 7 In the **Create External Networks** dialog box, in the **Name** field, enter a name for the external network.

Step 8 Expand **Subnet**.

Step 9 In the **Create Subnet** dialog box, perform the following actions:
a) In the **IP Address** field, enter the IP address.
b) In the Scope field, check the appropriate checkbox. Click **OK**.

Step 10 In the **Create External Network** dialog box, perform the following actions:
a) Expand **Subnet** to add another subnet.
b) In the **Create Subnet** dialog box, in the **IP Address** field, enter an IP address.
c) In the **Scope** field, check the appropriate check boxes. Click **OK**.
Note
- The import control policy is not enabled by default but can be enabled by the user. The import control policy is supported for BGP but not for EIGRP or for OSPF. If the user enables the import control policy for an unsupported protocol, it will be automatically ignored.
- The export control policy is supported for BGP, EIGRP, and OSPF.
- Route aggregation is also supported and the user can optionally choose route aggregation in the desired export or import direction. This feature is available for 0.0.0.0/0 and for the security option. If the import control policy is not enabled, an example of the check boxes to check are **Export Subnet**, **Security Import Subnet**, and **Aggregate Export**. The user must choose route map and security options.
- If an explicit route control policy is configured for a Layer 3 outside, then only specific Layer 3 outside policies are supported. Explicit route control policies are not supported for aggregate routes.

d) (Optional) In the **Route Summarization Policy** field, from the drop-down list, choose an existing route summarization policy or create a new one as desired and you must check the check box for **Export Route Control Subnet**.

e) In the **Create External Network** dialog box, click **OK**.

Step 11 In the **Create Routed Outside** dialog box, click **Finish**.

Step 12 In the **Navigation** pane, under ***Tenant_name*** > **Networking** > **Bridge Domains** and choose the *Bridge_Domain_name*.

Step 13 In the **Navigation** pane, choose the BD you created. In the **Work** pane, choose the **L3 Configurations** tab and in the **Associated L3 Outs** field, associate the desired L3 Out and choose the desired L3 Out for Route Profile. Click **Update**.
If the L3 Out is static, you are not required to choose any settings.

Step 14 **Note** To set attributes for BGP, OSPF, or EIGRP communication for all routes we receive, create default-import route control profile, create the appropriate set actions and no match actions.
In the **Navigation** pane, click **Route Profiles**, right-click **Create Route Profiles**, and in the **Create Route Profiles** dialog box, perform the following actions:

a) In the **Name** field, enter a name.

b) In the **Type** field, you must click **Use Routing Policy Only**. Click **Submit**.

Configuring Layer 3 Outside for Tenant Networks Using the REST API

The external routed network configured in the example can also be extended to support IPv4. Both IPv4 and IPv6 routes can be advertised to and learned from the external routed network.

Before You Begin

- The tenant, private network, and bridge domain are created.
- The external routed domain is created.

Procedure

Configure L3 Outside for tenant networks and associate the bridge domain with the Layer3 Outside.

Example:

```
<l3extOut name="L3Out1" enforceRtctrl="import,export">
            <l3extRsL3DomAtt tDn="uni/l3dom-l3DomP"/>
        <l3extLNodeP name="LNodeP1" >
            <l3extRsNodeL3OutAtt rtrId="1.2.3.4" tDn="topology/pod-1/node-101">
                <l3extLoopBackIfP addr="10.10.11.1" />
                <l3extLoopBackIfP addr="2000::3" />
            </l3extRsNodeL3OutAtt>
            <l3extLIfP  name="IFP1" >
                <l3extRsPathL3OutAtt addr="10.11.12.10/24"  ifInstT="l3-port"
tDn="topology/pod-1/paths-103/pathep-[eth1/17]" />
            </l3extLIfP>
            <l3extLIfP  name="IFP2" >
                <l3extRsPathL3OutAtt addr="2001::3/64"  ifInstT="l3-port"
tDn="topology/pod-1/paths-103/pathep-[eth1/17]" />
            </l3extLIfP>
        </l3extLNodeP>
        <l3extRsEctx tnFvCtxName="VRF1"/>
        <l3extInstP name="InstP1" >
            <l3extSubnet ip="192.168.1.0/24" scope="import-security" aggregate=""/>
           <l3extSubnet ip="0.0.0.0/0" scope="export-rtctrl,import-rtctrl,import-security"
```

```
 aggregate="export-rtctrl,import-rtctrl"/>
            <l3extSubnet ip="192.168.2.0/24" scope="export-rtctrl" aggregate=""/>
            <l3extSubnet ip="::/0" scope="import-rtctrl,import-security"
aggregate="import-rtctrl"/>
            <l3extSubnet ip="2001:17a::/64" scope="export-rtctrl" aggregate=""/>
        </l3extInstP>
    </l3extOut>
```

Note The "enforceRtctrl=import" is not applicable for OSPF and EIGRP.

Configuring a Layer 3 Outside for Tenant Networks Using the NX-OS Style CLI

The following steps describe how to configure a Layer 3 outside network for tenant networks as well as how to configure Layer 3 outside networks for individual protocols.

Before You Begin

Configure a tenant and VRF.

Procedure

Step 1 The following example shows how to deploy a node and L3 port for tenant VRF external L3 connectivity using the NX-OS CLI:

Example:

```
apic1# configure
apic1(config)# leaf 101
apic1(config-leaf)# vrf context tenant exampleCorp vrf v1
apic1(config-leaf-vrf)# router-id 1.2.3.4
apic1(config-leaf-vrf)# ip route 21.1.1.1/32 32.1.1.1
apic1(config-leaf-vrf)# ipv6 route 5001::1/128 6002::1 preferred
apic1(config-leaf-vrf)# exit

apic1(config-leaf)# interface eth 1/1
apic1(config-leaf-if)# vlan-domain member dom1
apic1(config-leaf-if)# no switchport
apic1(config-leaf-if)# vrf member tenant exampleCorp vrf v1
apic1(config-leaf-if)# ip address 10.1.1.1/24
apic1(config-leaf-if)# ip address 11.1.1.1/24 secondary
apic1(config-leaf-if)# ipv6 address 2001::1/64 preferred
apic1(config-leaf-if)# ipv6 link-local fe80::1
apic1(config-leaf-if)# mac-address 00:44:55:66:55::01
apic1(config-leaf-if)# mtu 4470
```

Step 2 The following shows how to configure a route map using the NX-OS CLI:

Example:

```
apic1(config)# leaf 101
apic1(config-leaf)# vrf context tenant exampleCorp vrf v1
apic1(config-leaf-vrf)# router-id 1.2.3.4
apic1(config-leaf-vrf)# route-map rtMap1
apic1(config-leaf-vrf)#scope global
apic1(config-leaf-vrf-route-map)# ip prefix-list list1 permit 13.13.13.0/24
apic1(config-leaf-vrf-route-map)# match prefix-list list1
apic1(config-leaf-vrf-route-map-match)# set metric 128
apic1(config-leaf-vrf-route-map-match)# set metric-type type-2
apic1(config-leaf-vrf-route-map-match)# set local-preference 64
apic1(config-leaf-vrf-route-map-match)# set community extended 20:22 additive
```

```
apic1(config-leaf-vrf-route-map-match)# set tag 1111
apic1(config-leaf-vrf-route-map-match)# contract provider prov1
apic1(config-leaf-vrf-route-map-match)# exit
apic1(config-leaf-vrf-route-map)# match bridge-domain bd1
```

Step 3 The following shows how to a configure Layer 3 outside network for the BGP protocol:

Example:

```
apic1# configure
apic1(config)# leaf 101
apic1(config-leaf)# router bgp 100
apic1(config-bgp)# vrf member tenant exampleCorp vrf v100
apic1(config-leaf-bgp-vrf)# neighbor 192.0.2.10/32
apic1(config-leaf-bgp-vrf)# neighbor 192.0.2.11/32
apic1(config-leaf-bgp-vrf-neighbor)# address-family ipv4 unicast
apic1(config-leaf-bgp-vrf-neighbor-af)# maximum-prefix 10 threshold 10 action restart
restart-time 10
apic1(config-leaf-bgp-vrf-neighbor-af)# exit
apic1(config-leaf-bgp-vrf-neighbor)# remote-as 200
apic1(config-leaf-bgp-vrf-neighbor)# update-source ethernet 1/1
apic1(config-leaf-bgp-vrf-neighbor)# route-map rtMap1 out
apic1(config-leaf-bgp-vrf-neighbor)# exit

To configure route-summarization
apic1(config)# leaf 101
apic1(config-leaf)# router bgp 100
apic1(config-bgp)# vrf member tenant exampleCorp vrf v100
apic1(config-leaf-bgp-vrf)# aggregate-address 192.0.2.0/28 as-set
```

Step 4 The following shows how to configure a Layer 3 outside network for the OSPF protocol:

Example:

```
apic1# configure
apic1(config)# leaf 102
apic1(config-leaf)# router ospf default
apic1(config-leaf-ospf)# vrf member tenant exampleCorp vrf v100
apic1(config-leaf-ospf-vrf)# area 0 nssa
apic1(config-leaf-ospf-vrf)# area 17 stub
apic1(config-leaf-ospf-vrf)# area 17 default-cost 20
apic1(config-leaf-ospf-vrf)# area 17 route-map ospf-to-eigrp out
apic1(config-leaf-ospf-vrf)# area 17 loopback 192.0.20.11/32
apic1(config-leaf-ospf-vrf)# inherit ipv4 rtMap1 vrfTemplate2
apic1(config-leaf-ospf-vrf)# exit
apic1(config-leaf-ospf)# exit
apic1(config-leaf)# interface eth 1/3
apic1(config-leaf-if)# ip router ospf default area 17
apic1(config-leaf-if)# ip ospf inherit interface-policy ifPolicy3 tenant exampleCorp
apic1(config-leaf-if)# ip ospf authentication md5
apic1(config-leaf-if)# ip ospf authentication-key c1$c0123
```

a) The following shows how to configure OSPF External Route Summarization:

Example:

```
apic1(config)# leaf 101
apic1(config-leaf)# router ospf default
apic1(config-leaf-ospf)# vrf member tenant exampleCorp vrf v100
apic1(config-leaf-ospf-vrf)# summary-address 182.1.20.0/24
```

b) The following shows how to configure OSPF Inter-Area Summarization, which is used to summarize networks between areas:

Example:

```
apic1(config-leaf-ospf-vrf)# area 17 range 192.0.20.0/24 cost 20
```

Step 5 The following shows how to configure a Layer 3 outside network for the EIGRP protocol:

Example:

```
apic1# configure
apic1(config)# leaf 101
apic1(config-leaf)# router eigrp default
apic1(config-eigrp)# vrf member tenant exampleCorp vrf v100
apic1(config-eigrp-vrf)# autonomous-system 300
apic1(config-eigrp-vrf)# exit
apic1(config-eigrp)# exit

apic1(config-leaf)# interface ethernet 1/21
apic1(config-leaf-if)# no switchport
5
apic1(config-leaf-if)# vlan-domain member dom1
apic1(config-leaf-if)# vrf member tenant exampleCorp vrf v100
apic1(config-leaf-if)# ip address 181.12.12.1/24
apic1(config-leaf-if)# ip router eigrp default
apic1(config-leaf-if)# ip distribute-list eigrp default route-map rMapT5 out
distribute list will be updated on all EIGRP interfaces on node 1021 VRF exampleCorp/v100
apic1(config-leaf-if)# ip hello-interval eigrp default 5
apic1(config-leaf-if)# ip hold-interval eigrp default 10
apic1(config-leaf-if)# ip next-hop-self eigrp default
apic1(config-leaf-if)# ip passive-interface eigrp default
apic1(config-leaf-if)# ip split-horizon eigrp default
apic1(config-leaf-if)# inherit eigrp ip interface-policy ifTemplate5
```

a) The following shows how to configure EIGRP summarization:

Example:

```
apic1(config-leaf-if)# ip summary-address eigrp default 172.10.1.0/24
apic1(config-leaf-if)# exit
apic1(config-leaf)# exit
apic1(config)# exit
```

Step 6 The following shows how to configure an external-L3 EPG and deploy it on a leaf switch:

Example:

```
apic1# configure
apic1(config)# tenant exampleCorp
# CONFIGURE EXTERNAL L3 EPG
apic1(config-tenant)# external-l3 epg epgExtern1
apic1(config-tenant-l3ext-epg)# vrf member v1
apic1(config-tenant-l3ext-epg)# match ip 192.0.20.0/24
apic1(config-tenant-l3ext-epg)# match ipv6 2001::1/64
apic1(config-tenant-l3ext-epg)# set qos-class level1
apic1(config-tenant-l3ext-epg)# set dscp af31
apic1(config-tenant-l3ext-epg)# contract consumer cConsumer1
apic1(config-tenant-l3ext-epg)# contract provider cProvider1
apic1(config-tenant-l3ext-epg)# contract deny cDeny1
apic1(config-tenant-l3ext-epg)# exit
apic1(config-tenant)# exit
# DEPLOY EXTERNAL L3 EPG ON A LEAF
apic1(config)# leaf 101
```

```
apic1(config-leaf)# vrf context tenant exampleCorp vrf
```

Shared Services Contracts Usage

Shared services enable communications across tenants while preserving the isolation and security policies of the tenants. A routed connection to an external network is an example of a shared service that multiple tenants use.

Follow these guidelines when configuring shared services contracts.

- For shared services that export subnets to different contexts (VRFs), the subnet must be configured under an EPG, and the scope must be set to *advertised externally* and *shared between VRFs*.
- Contracts are not needed for inter-bridge domain traffic when a private network is unenforced.
- Contracts are needed for shared service inter-context (VRF) traffic even when a context (VRF) is unenforced.
- The context (VRF) of a provider EPG cannot be in unenforced mode while providing a shared service.
- A shared service is supported only with non-overlapping and non-duplicate subnets. When configuring subnets for shared services, follow these guidelines:
  - Configure the subnet for a shared service provider under the EPG, not under the bridge domain.
  - Subnets configured under an EPG that share the same context must be disjointed and must not overlap.
  - Subnets leaked from one context to another must be disjointed and must not overlap.
  - Subnets leaked from multiple consumer networks into a context or vice versa must be disjointed and must not overlap.

Note If two consumers are mistakenly configured with the same subnet, recover from this condition by removing the subnet configuration for both then reconfigure the subnets correctly.

- Do not configure a shared service with `AnyToProv` in the provider context. The APIC rejects this configuration and raises a fault.
- When a contract is configured between in-band and out-of-band EPGs, the following restrictions apply:
  - Both EPGs should be in the same context (VRF).
  - Ffilters apply in the incoming direction only.
  - Layer 2 filters are not supported.
  - QoS does not apply to in-band Layer 4 to Layer 7 services.
  - Management statistics are not available.

- Shared services for CPU-bound traffic are not supported.

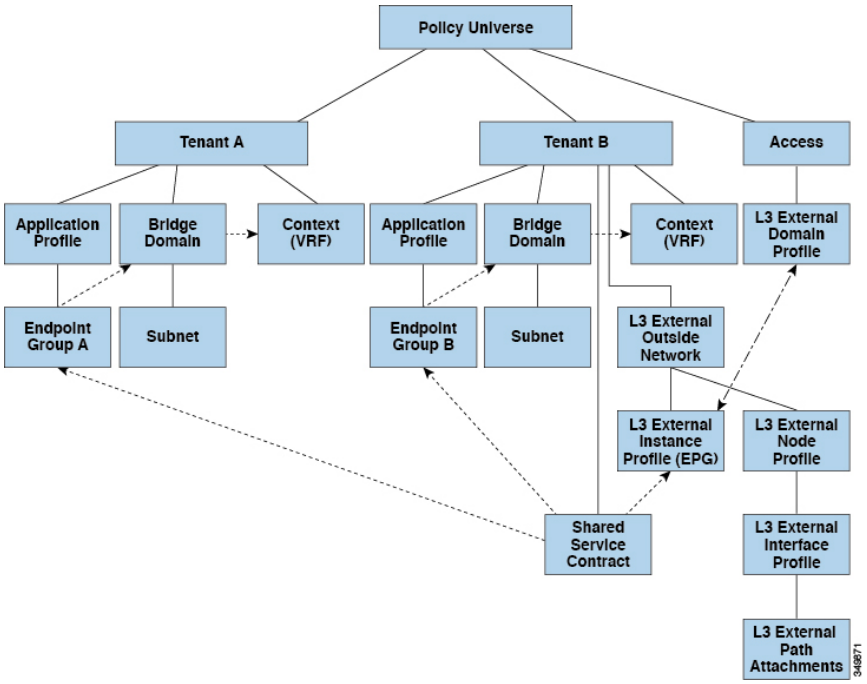
Shared Layer 3 Out

A shared Layer 3 Out configuration provides routed connectivity to external networks as a shared service. An `l3extInstP` EPG provides routed connectivity to external networks. It can be can be provisioned as a shared service in any tenant (user, common, infra, or mgmt.). Prior to release 1.2(1x), this configuration was only supported in the user and common tenants. An EPG in any tenant can use a shared services contract to connect with an `l3extInstP` EPG regardless of where in the fabric that `l3extInstP` EPG is provisioned. This simplifies the provisioning of routed connectivity to external networks; multiple tenants can share a single `l3extInstP` EPG for routed connectivity to external networks. Sharing an `l3extInstP` EPG is more efficient because it consumes only one session on the switch regardless of how many EPGs use the single shared `l3extInstP` EPG.

Note All switches that will use l3extInstP EPG shared service contracts require the hardware and software support available starting with the APIC 1.2(1x) and switch 11.2(1x) releases. Refer to the Firmware Management Guide and Release Notes documentation for more details.

The figure below illustrates the major policy model objects that are configured for a shared `l3extInstP` EPG.

Figure 26: Shared Layer 3 Out Policy Model



Take note of the following guidelines and limitations for shared Layer 3 Out configurations::

- No tenant limitations: Tenants A and B can be any kind of tenant (user, common, infra, mgmt.). The shared l3extInstP EPG does not have to be in tenant common.
- Flexible placement of EPGs: EPG A and EPG B in the illustration above are in different tenants. EPG A and EPG B could use the same bridge domain and context, but they are not required to do so. EPG A and EPG B are in different bridge domains and different contexts but still share the same `l3extInstP` EPG.
- A subnet can be *private*, *public*, or *shared*. A subnet that is to be leaked into a consumer or provider EPG of a Layer 3 External Outside Network must be set to *shared*. A subnet that is to be exported to a Layer 3 External Outside Network must be set to *public*.
- The shared service contract is exported from the tenant that contains the `l3extInstP` EPG that provides shared Layer 3 Out service. The shared service contract is imported into the tenants that contain the EPGs that consume the shared service.
- Do not use taboo contracts with a shared L3 out; this configuration is not supported.
- The `l3extInstP` as a shared service provider is supported, but only with non l3extInstP consumers (where Layer3Out EPg = l3extInstP).
- Transit routing is not supported with shared services. In other words, two Layer3 Outs in different VRFs cannot communicate with each other using the shared services feature.
- Traffic Flap: When an `l3instP` EPG is configured with an external subnet of 0.0.0.0/0 with the scope property of the `l3instP` subnet set to shared route control (*shared-rctrl*), or shared security (*shared-security*), the context (VRF) is redeployed with a global `pcTag`. This will flap all the external traffic in that VRF (because the VRF is redeployed with a global `pcTag`).
- Prefixes for a shared Layer 3 out must to be unique. Multiple shared Layer 3 Out configurations with the same prefix in the same context (VRF) will not work. Be sure that the external subnets (external prefixes) getting leaked into a VRF are be unique (the same external subnet cannot belong to multiple `l3instPs`). A Layer 3 outside configuration (for example, named L3Out1) with prefix1 and a second Layer 3 outside configuration (for example, named L3Out2) also with prefix1 belonging to the same context (VRF) will not work (because only 1 pcTag will be deployed).
- Traffic not permitted: Traffic is not permitted when an invalid configuration sets the scope of the external subnet to shared route control (shared-rtctrl) as a subset of a subnet that is set to shared security(shared-security). For example, the following configuration is invalid:
    - *shared rtctrl*: 10.1.1.0/24, 10.1.2.0/24
    - *shared security*: 10.1.0.0/16

  In this case, traffic coming in on a non-border leaf with a destination IP of 10.1.1.1 will get dropped since prefixes 10.1.1.0/24 and 10.1.2.0/24 are installed with a drop rule. Traffic is not permitted. Such traffic can be enabled by revising the configuration to use the shared-rtctrl prefixes as shared-security prefixes as well.
- Inadvertent traffic flow: Prevent inadvertent traffic flow by avoiding the following configuration scenarios:
    - **Case 1** configuration details:
        - A Layer 3 outside configuration (for example, named L3Out1) with context (VRF) 1 is called provider1.
        - A second Layer 3 outside configuration (for example, named L3Out2) with context (VRF) 2 is called provider2.

- L3Out1 VRF1 advertises a default route to the Internet = 0.0.0.0/0 = *shared-rtctrl*, *shared-security*.
- L3Out2 VRF2 advertises specific subnets to DNS and NTP = 192.0.0.0/8 = *shared-rtctrl*.
- L3Out2 VRF2 has specific subnets 192.1.0.0/16 = *shared-security*.
- **Variation A**: EPG Traffic Goes to Multiple Contexts (VRFs).
  - Communications between EPG1 and L3Out1 is regulated by an *allow_all* contract.
  - Communications between EPG1 and L3Out2 is regulated by an *allow_all* contract.

  Result: Traffic from EPG1 to L3Out2 also goes to 192.2.x.x.
- **Variation B**: An EPG conforms to *allow_all* contract of second shared Layer 3 out.
  - Communications between EPG1 and L3Out1 is regulated by an *allow_all* contract.
  - Communications between EPG1 and L3Out2 is regulated by an *allow_icmp* contract.

  Result: Traffic from EPG1 to L3Out2 to 192.2.x.x conforms to the *allow_all* contract.

- **Case 2** configuration details:
  - A Layer 3 out instance profile (l3instP) has one shared prefix and other non-shared prefixes.
  - Traffic coming in with `src = non-shared` is allowed to go to the EPG
    - **Variation A**: Unintended traffic goes through an EPG.
      - Layer 3 out (l3instP) EPG traffic goes through a Layer 3 out that has these prefixes:
        - `192.0.0.0/8 = import-security, shared-rtctrl`
        - `192.1.0.0/16 = shared-security`
        - The EPG has `1.1.0.0/16 = shared`

        Result: Traffic going from 192.2.x.x also goes through to the EPG.
    - **Variation B**: Unintended traffic goes through an EPG. Traffic coming in a shared layer 3 out can go through if the context (VRF) has an .
      - The shared Layer 3 out context (VRF) has an EPG with `pcTag = prov vrf` and a contract that is *allow_all*
      - The EPG `<subnet> = shared`.

      Result: The traffic coming in on the Layer 3 out can go through the EPG.

Neighbor Discovery

The IPv6 Neighbor Discovery (ND) protocol is responsible for address autoconfiguration of nodes, discovery of other nodes on the link, determining the link-layer addresses of other nodes, duplicate address detection,

finding available routers and DNS servers, address prefix discovery, and maintaining reachability information about the paths to other active neighbor nodes.

ND-specific Neighbor Solicitation/Neighbor Advertisement (NS/NA) and Router Solicitation/Router Advertisement (RS/RA) packet types are supported on all ACI fabric Layer 3 interfaces, including physical, L3 Sub-if, and SVI (external and pervasive). RS/RA packets are used for autoconfiguration for all L3 interfaces but are only configurable for pervasive SVIs. ACI bridge domain ND always operates in flood mode; unicast mode is not supported.

The ACI fabric ND support includes the following:

- Interface policies (`nd:IfPol`) control ND timers and behavior for NS/NA messages.
- ND prefix policies (`nd:PfxPol`) controls RA messages.
- Configuration of IPv6 subnets for ND (fv:Subnet).
- ND interface policies for external networks.
- Configurable ND subnets for external networks, and arbitrary subnet configurations for pervasive bridge domains are not supported.

Configuration options include the following:

- Adjacencies
  - Configurable Static Adjacencies : (<vrf, L3Iface, ipv6 address> --> mac address)
  - Dynamic Adjacencies : Learnt via exchange of NS/NA packets
- Per Interface
  - Control of ND packets (NS/NA)
    - Neighbor Solicitation Interval
    - Neighbor Solicitation Retry count
  - Control of RA packets
    - Suppress RA
    - Suppress RA MTU
    - RA Interval, RA Interval minimum, Retransmit time
- Per Prefix (advertised in RAs) control
  - Lifetime, preferred lifetime
  - Prefix Control (autoconfiguration, on link)

Creating the Tenant, VRF, and Bridge Domain with IPv6 Neighbor Discovery Using the Advanced GUI

This task shows how to create a tenant, a VRF, and a bridge domain (BD) within which two different types of Neighbor Discovery (ND) policies are created. They are ND interface policy and ND prefix policy. While ND interface policies are deployed under BDs, ND prefix policies are deployed for individual subnets. Each BD can have its own ND interface policy . The ND interface policy is deployed on all IPv6 interfaces by default. In Cisco APIC, there is already an ND interface default policy available to use. If desired, you can create a custom ND interface policy to use instead. The ND prefix policy is on a subnet level. Every BD can have multiple subnets, and each subnet can have a different ND prefix policy.

Procedure

Step 1 On the menu bar, click **TENANT** > **Add Tenant**.

Step 2 In the **Create Tenant** dialog box, perform the following tasks:

a) in the **Name** field, enter a name.
b) Click the **Security Domains** + icon to open the **Create Security Domain** dialog box.
c) In the **Name** field, enter a name for the security domain. Click **Submit**.
d) In the **Create Tenant** dialog box, check the check box for the security domain that you created, and click **Submit**.

Step 3 In the **Navigation** pane, expand **Tenant-name** > **Networking**. In the **Work** pane, drag the **VRF** icon to the canvas to open the **Create VRF** dialog box, and perform the following actions:

a) In the **Name** field, enter a name.
b) Click **Submit** to complete the **VRF** configuration.

Step 4 In the **Networking** area, drag the **BD** icon to the canvas while connecting it to the **VRF** icon. In the **Create Bridge Domain** dialog box that displays, perform the following actions:

a) In the **Name** field, enter a name.
b) Click the **L3 Configurations** tab, and expand **Subnets** to open the **Create Subnet** dialog box, enter the subnet mask in the **Gateway IP** field.

Step 5 In the **Subnet Control** field, ensure that the **ND RA Prefix** check box is checked.

Step 6 In the **ND Prefix policy** field drop-down list, click **Create ND RA Prefix Policy**.

Note There is already a default policy available that will be deployed on all IPv6 interfaces. Alternatively, you can create an ND prefix policy to use as shown in this example. By default, the IPv6 gateway subnets are advertised as ND prefixes in the ND RA messages. A user can choose to not advertise the subnet in ND RA messages by un-checking the ND RA prefix check box.

Step 7 In the **Create ND RA Prefix Policy** dialog box, perform the following actions:

a) In the **Name** field, enter the name for the prefix policy.
Note For a given subnet there can only be one prefix policy. It is possible for each subnet to have a different prefix policy, although subnets can use a common prefix policy.
b) In the **Controller State** field, check the desired check boxes.
c) In the **Valid Prefix Lifetime** field, choose the desired value for how long you want the prefix to be valid.
d) In the **Preferred Prefix Lifetime** field, choose a desired value. Click **OK**.
Note An ND prefix policy is created and attached to the specific subnet.

Step 8 In the **ND policy** field drop-down list, click **Create ND Interface Policy** and perform the following tasks:

a) In the **Name** field, enter a name for the policy.

b) Click **Submit**.

Step 9 Click **OK** to complete the bridge domain configuration.
Similarly you can create additional subnets with different prefix policies as required.

A subnet with an IPv6 address is created under the BD and an ND prefix policy has been associated with it.

Creating the Tenant, VRF, and Bridge Domain with IPv6 Neighbor Discovery Using the REST API

Procedure

Create a tenant, VRF, bridge domain with a neighbor discovery interface policy and a neighbor discovery prefix policy.

Example:

```
<fvTenant descr="" dn="uni/tn-ExampleCorp" name="ExampleCorp" ownerKey="" ownerTag="">
    <ndIfPol name="NDPol001" ctrl="managed-cfg"  descr="" hopLimit="64" mtu="1500"
nsIntvl="1000" nsRetries="3" ownerKey="" ownerTag="" raIntvl="600" raLifetime="1800"
reachableTime="0" retransTimer="0"/>
    <fvCtx descr="" knwMcastAct="permit" name="pvn1" ownerKey="" ownerTag=""
pcEnfPref="enforced">
    </fvCtx>
   <fvBD arpFlood="no" descr="" mac="00:22:BD:F8:19:FF" multiDstPktAct="bd-flood" name="bd1"
 ownerKey="" ownerTag="" unicastRoute="yes" unkMacUcastAct="proxy" unkMcastAct="flood">
        <fvRsBDToNdP tnNdIfPolName="NDPol001"/>
        <fvRsCtx tnFvCtxName="pvn1"/>
        <fvSubnet ctrl="nd" descr="" ip="34::1/64" name="" preferred="no" scope="private">
            <fvRsNdPfxPol tnNdPfxPolName="NDPfxPol001"/>
        </fvSubnet>
        <fvSubnet ctrl="nd" descr="" ip="33::1/64" name="" preferred="no" scope="private">
            <fvRsNdPfxPol tnNdPfxPolName="NDPfxPol002"/>
        </fvSubnet>
    </fvBD>
   <ndPfxPol ctrl="auto-cfg,on-link" descr="" lifetime="1000" name="NDPfxPol001" ownerKey=""
 ownerTag="" prefLifetime="1000"/>
    <ndPfxPol ctrl="auto-cfg,on-link" descr="" lifetime="4294967295" name="NDPfxPol002"
ownerKey="" ownerTag="" prefLifetime="4294967295"/>
</fvTenant>
```

Note If you have a public subnet when you configure the routed outside, you must associate the bridge domain with the outside configuration.

Configuring a Tenant, VRF, and Bridge Domain with IPv6 Neighbor Discovery Using the CLI

Procedure

Step 1 In the CLI, change the directory to `/aci`.

Example:

```
admin@apic1:~> cd /aci
```

Step 2 Configure a neighbor discovery interface policy.

Example:

```
admin@apic1:aci> cd tenants/
admin@apic1:tenants> mocreate ExampleCorp
admin@apic1:tenants> moconfig commit
admin@apic1:tenants> cd ExampleCorp/
admin@apic1:ExampleCorp> cd networking/protocol-policies/nd/
admin@apic1:nd> mocreate interface-policy NDPol001
admin@apic1:nd> moconfig commit
admin@apic1:nd> cd interface-policy-NDPol001/
admin@apic1:interface-policy-NDPol001> moset mtu 1500
admin@apic1:interface-policy-NDPol001> moconfig commit
admin@apic1:interface-policy-NDPol001>
admin@apic1:interface-policy-NDPol001> cd ../../../private-networks/
admin@apic1:private-networks> mocreate pvn1
admin@apic1:private-networks> moconfig commit
admin@apic1:pvn1> cd ../../bridge-domains/
admin@apic1:bridge-domains> mocreate bd1
admin@apic1:bridge-domains> cd bd1
admin@apic1:bd1> moset custom-mac-address 00:22:BD:F8:19:FF
admin@apic1:bd1> moset nd-interface-policy NDPol001
admin@apic1:bd1> moconfig commit
```

Step 3 Configure a neighbor discovery prefix policy.

Example:

```
admin@apic1:bd1> cd ../../protocol-policies/nd/
admin@apic1:nd> mocreate prefix-policy NDPfxPol001
admin@apic1:nd> cd prefix-policy-NDPfxPol001/
admin@apic1:prefix-policy-NDPfxPol001> moset valid-lifetime 1000
admin@apic1:prefix-policy-NDPfxPol001> moset preferred-lifetime 1000
admin@apic1:prefix-policy-NDPfxPol001> moconfig commit
admin@apic1:prefix-policy-NDPfxPol001> cd ../
admin@apic1:nd> mocreate prefix-policy NDPfxPol002
admin@apic1:nd> cd prefix-policy-NDPfxPol002/
admin@apic1:prefix-policy-NDPfxPol002> moset valid-lifetime 4294967295
admin@apic1:prefix-policy-NDPfxPol002> moset preferred-lifetime 4294967295
admin@apic1:prefix-policy-NDPfxPol002> moconfig commit
admin@apic1:prefix-policy-NDPfxPol002> cd ../../../bridge-domains/bd1/subnets/
admin@apic1:subnets> mocreate 34::1/64
admin@apic1:subnets> cd 34::1_64/
admin@apic1:34::1_64> moset nd-prefix-policy NDPfxPol001
admin@apic1:34::1_64> moconfig commit
admin@apic1:34::1_64> cd ../
admin@apic1:subnets> mocreate 33::1/64
admin@apic1:subnets> cd 33::1_64/
```

```
admin@apic1:33::1_64> moset nd-prefix-policy NDPfxPol002
admin@apic1:33::1_64> moconfig commit
```

Configuring a Routing Control Protocol Using Import and Export Controls

This topic provides a typical example of how to configure a routing control protocol using import and export controls when using Cisco APIC.

Note When you configure Layer 3 Outside (L3Out) connections to external routers, it is critical that the MTU be set appropriately on both sides. On some platforms, such as ACI, Cisco NX-OS, and Cisco IOS, the configurable MTU value takes into account packet headers (resulting in a max packet size to be set as 9000 bytes), whereas other platforms such as IOS-XR configure the MTU value exclusive of packet headers (resulting in a max packet size of 8986 bytes). For the appropriate MTU values for each platform, see the relevant configuration guides. Cisco highly recommends you test the MTU using CLI-based commands. For example, on the Cisco NX-OS CLI, use a command such as `ping 1.1.1.1 df-bit packet-size 9000 source-interface ethernet 1/1`

Configuring a Route Control Protocol to Use Import and Export Controls Using the GUI

This task lists steps to create import and export policies. By default, import controls are not enforced, so the import control must be manually assigned.

Before You Begin

- The tenant, private network, and bridge domain are created.
- The Layer 3 outside for tenant networks is created.

Procedure

Step 1 On the menu bar, click **TENANTS** > ***Tenant_name*** > **Networking** > **External Routed Networks** > ***Layer3_Outside_name*** .

Step 2 Right click *Layer3_Outside_name* and click **Create Route Profile**.

Step 3 In the **Create Route Profile** dialog box, perform the following actions:

a) From the **Name** field drop-down list, choose the appropriate route profile.
Depending on your selection, whatever is advertised on the specific outside is automatically used.
b) In the **Type** field, choose **Combining Subnets with Routing Policy**.
c) Expand **Order**.

Step 4 In the **Create Route Control Context** dialog box, perform the following actions:

a) In the **Order** field, choose the desired order number.
b) In the **Name** field, enter a name for the route control private network.
c) From the **Match Rule** field drop-down list, click **Create Match Rule**.
d) In the **Create Match Rule** dialog box, in the **Name** field, enter a route match rule name. Click **Submit**. Specify the match community regular expression term and match community terms as desired. Match community factors will require you to specify the name, community and scope.
e) From the **Set Attribute** drop-down list, choose **Create Action Rule Profile**.
f) In the **Create Action Rule Profile** dialog box, in the **Name** field, enter a name for the rule.
g) Check the check boxes for the desired rules you want to set, and choose the appropriate values that are displayed for the choices. Click **Submit**.
The policy is created and associated with the action rule.
h) Click **OK**.
i) In the **Create Route Profile** dialog box, click **Submit**.

Step 5 In the **Navigation** pane, choose **Route Profile** > ***route_profile_name*** > ***route_control_private_network_name***.
In the **Work** pane, under **Properties** the route profile policy and the associated action rule name are displayed.

Step 6 In the **Navigation** pane, click the ***Layer3_Outside_name***.
In the Work pane, the Properties are displayed.

Step 7 (Optional) Click the **Route Control Enforcement** field and enter **Import Control** to enable the import policy.
The import control policy is not enabled by default but can be enabled by the user. The import control policy is supported for BGP but not for EIGRP or for OSPF. If the user enables the import control policy for an unsupported protocol, it will be automatically ignored. The export control policy is supported for BGP, EIGRP, and OSPF.

Step 8 To create a customized export policy, right-click **Route Profiles**, click **Create Route Profiles**, and perform the following actions:

a) In the **Create Route Profile** dialog box, from the drop-down list in the **Name** field, choose a name for the export policy.
b) Expand the + sign in the dialog box.
c) In the **Create Route Control Context** dialog box, in the **Order** field, choose a value.
d) In the **Name** field, enter a name for the route control private network.
e) (Optional) From the **Match Rule** field drop-down list, choose **Create Route Control Context**, and create and attach a match rule policy if desired.
f) From the **Set Attribute** field drop-down list, choose **Create Action Rule Profile**.
Alternatively, if desired, you can choose an existing set action, and click **Submit**.
g) In the **Create Action Rule Profile** dialog box, in the **Name** field, enter a name.
h) Check the check boxes for the desired rules you want to set, and choose the appropriate values that are displayed for the choices. Click **Submit**.
In the **Create Route Control Context** dialog box, the policy is created and associated with the action rule.
i) Click **OK**.
j) In the **Create Route Profile** dialog box, click **Submit**.

In the **Work** pane, the export policy is displayed.

Note To enable the export policy, it must first be applied. For the purpose of this example, it is applied to all the subnets under the network.

Step 9 In the **Navigation** pane, expand **External Routed Networks** > ***External_Routed_Network_name*** > **Networks** > ***Network_name***, and perform the following actions:

a) From the **Name** field drop-down list, choose the policy created earlier.
b) In the **Direction** field, from the drop-down list, choose **Route Control Profile**. Click **Update**.

Step 10 Click **Submit**.

Configuring a Route Control Protocol to Use Import and Export Controls Using the REST API

Before You Begin

- The tenant, private network, and bridge domain are created.
- The Layer 3 outside tenant network is configured.

Procedure

Configure the route control protocol using import and export controls.

Example:

```
<l3extOut descr="" dn="uni/tn-Ten_ND/out-L3Out1" enforceRtctrl="export" name="L3Out1"
ownerKey="" ownerTag="" targetDscp="unspecified">
        <l3extLNodeP descr="" name="LNodeP1" ownerKey="" ownerTag="" tag="yellow-green"
targetDscp="unspecified">
            <l3extRsNodeL3OutAtt rtrId="1.2.3.4" rtrIdLoopBack="yes"
tDn="topology/pod-1/node-101">
                <l3extLoopBackIfP addr="2000::3" descr="" name=""/>
            </l3extRsNodeL3OutAtt>
            <l3extLIfP descr="" name="IFP1" ownerKey="" ownerTag="" tag="yellow-green">
                <ospfIfP authKeyId="1" authType="none" descr="" name="">
                    <ospfRsIfPol tnOspfIfPolName=""/>
                </ospfIfP>
                <l3extRsNdIfPol tnNdIfPolName=""/>
                <l3extRsPathL3OutAtt addr="10.11.12.10/24" descr="" encap="unknown"
ifInstT="l3-port"
llAddr="::" mac="00:22:BD:F8:19:FF" mtu="1500" tDn="topology/pod-1/paths-101/pathep-[eth1/17]"
 targetDscp="unspecified"/>
            </l3extLIfP>
        </l3extLNodeP>
        <l3extRsEctx tnFvCtxName="PVN1"/>
        <l3extInstP descr="" matchT="AtleastOne" name="InstP1" prio="unspecified"
targetDscp="unspecified">
            <fvRsCustQosPol tnQosCustomPolName=""/>
            <l3extSubnet aggregate="" descr="" ip="192.168.1.0/24" name="" scope=""/>
        </l3extInstP>
        <ospfExtP areaCost="1" areaCtrl="redistribute,summary" areaId="0.0.0.1"
areaType="nssa" descr=""/>
        <rtctrlProfile descr="" name="default-export" ownerKey="" ownerTag="">
            <rtctrlCtxP descr="" name="routecontrolpvtnw" order="3">
                <rtctrlScope descr="" name="">
                    <rtctrlRsScopeToAttrP tnRtctrlAttrPName="actionruleprofile2"/>
                </rtctrlScope>
            </rtctrlCtxP>
        </rtctrlProfile>
    </l3extOut>
```

Configuring Route Control Protocol Using Import and Export Controls Using the NX-OS Style CLI

This section describes how to create a route map using the NX-OS CLI:

Before You Begin

- The tenant, private network, and bridge domain are created.
- The Layer 3 outside tenant network is configured.

Procedure

Step 1 Import Route control using match community, match prefix-list

Example:

```
apic1# configure
apic1(config)# leaf 101
    # Create community-list
apic1(config-leaf)# template community-list standard CL_1 65536:20 tenant exampleCorp
apic1(config-leaf)# vrf context tenant exampleCorp vrf v1

     #Create Route-map and use it for BGP import control.
apic1(config-leaf-vrf)# route-map bgpMap
    # Match prefix-list and set route-profile actions for the match.
apic1(config-leaf-vrf-route-map)# ip prefix-list list1 permit 13.13.13.0/24
apic1(config-leaf-vrf-route-map)# ip prefix-list list1 permit 14.14.14.0/24
apic1(config-leaf-vrf-route-map)# match prefix-list list1
apic1(config-leaf-vrf-route-map-match)# set tag 200
apic1(config-leaf-vrf-route-map-match)# set local-preference 64
    # Match community-list and set route-profile actions for the match.
apic1(config-leaf-vrf-route-map)# match community CL_1
apic1(config-leaf-vrf-route-map-match)# set metric 200
apic1(config-leaf-vrf-route-map-match)# set community extended 20:22 additive
     #Adding the route-map the protocol.
apic1(config-leaf)# router bgp 100
apic1(config-bgp)# vrf member tenant exampleCorp vrf v1
apic1(config-leaf-bgp-vrf)# neighbor 3.3.3.3
apic1(config-leaf-bgp-vrf-neighbor)# route-map bgpMap in
```

Step 2 Export Route Control using match BD, default-export route-profile

Example:

```
# Create custom and "default-export" route-profiles
apic1(config)# leaf 101
apic1(config-leaf)# vrf context tenant exampleCorp vrf v1
apic1(config-leaf-vrf)# template route-profile default-export
apic1(config-leaf-vrf-template-route-profile)# set metric 256
apic1(config-leaf-vrf)# template route-profile bd-rtctrl
apic1(config-leaf-vrf-template-route-profile)# set metric 128

#Create a Route-map and match on BD, prefix-list
apic1(config-leaf-vrf)# route-map bgpMap
apic1(config-leaf-vrf-route-map)# match bridge-domain bd1
apic1(config-leaf-vrf-route-map-match)#exit
apic1(config-leaf-vrf-route-map)# match prefix-list p1
apic1(config-leaf-vrf-route-map-match)#exit
apic1(config-leaf-vrf-route-map)# match bridge-domain bd2
```

```
apic1(config-leaf-vrf-route-map-match)# inherit route-profile bd-rtctrl
```

Note In this case, public-subnets from bd1 and prefixes matching prefix-list p1 are exported out using route-profile “default-export”, while public-subnets from bd2 are exported out using route-profile “bd-rtctrl”.

ACI Transit Routing

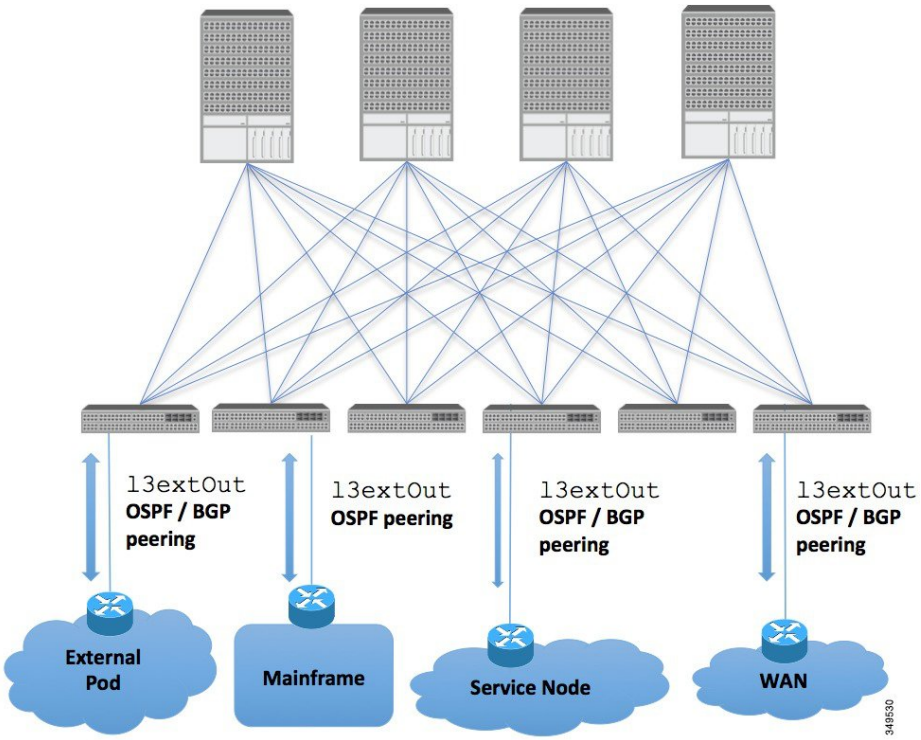
The ACI fabric supports transit routing, which enables border routers to perform bidirectional redistribution with other routing domains. Unlike the stub routing domains of earlier releases of the ACI Fabric that block transit redistribution, bidirectional redistribution passes routing information from one routing domain to another. Such redistribution lets the ACI fabric provide full IP connectivity between different routing domains. Doing so can also provide redundant connectivity by enabling backup paths between routing domains.

Design transit redistribution policies that avoid sub-optimal routing or the more serious problem of routing loops. Typically, transit redistribution does not preserve the original topology and link-state information and redistributes external routes in distance-vector fashion (routes are advertised as vector prefixes and associated distances even with link-state protocols). Under these circumstances, the routers can inadvertently form routing loops that fail to deliver packets to their destination.

Transit Routing Use Cases

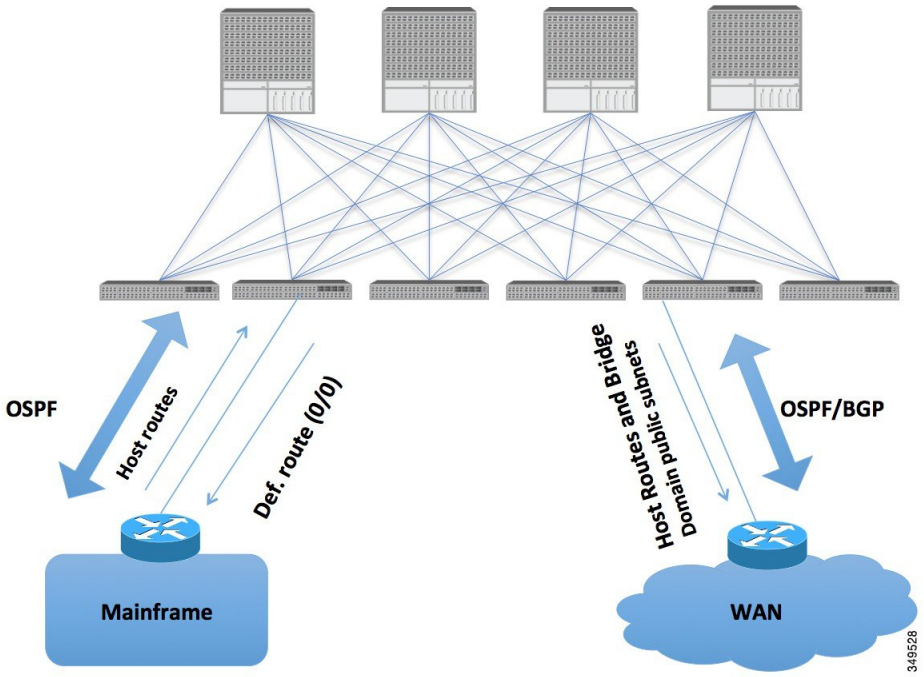
Multiple Layer 3 domains such as external pods, mainframes, service nodes, or WAN routers can peer with the ACI fabric to provide transit functionality between them.

Figure 27: Transit Routing between Layer 3 Domains



Mainframes can function as IP servers running standard IP routing protocols that accommodate requirements from Logical Partitions (LPARs) and Virtual IP Addressing (VIPA).

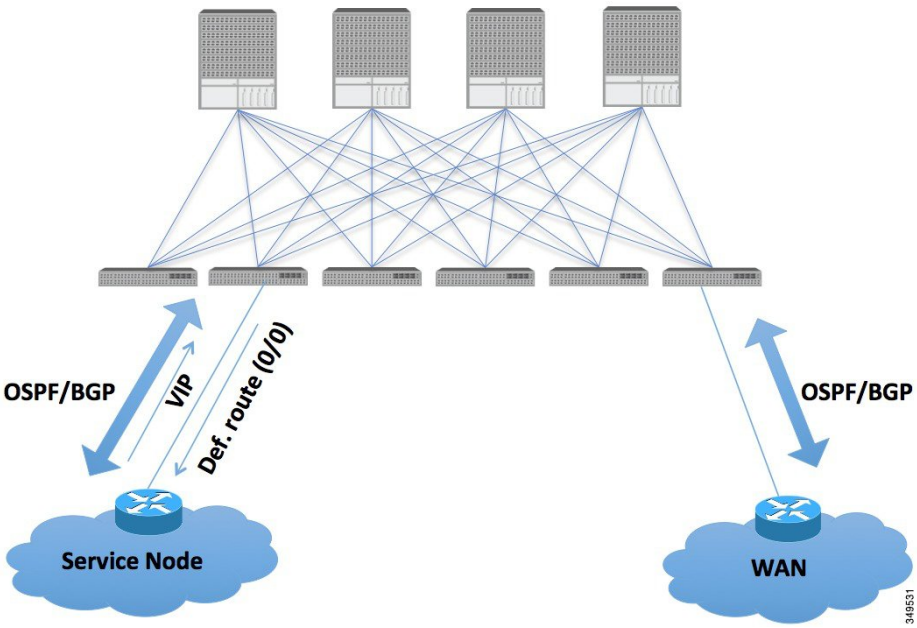
Figure 28: Mainframe Transit Connectivity



Mainframes that require the ACI fabric to be a transit domain for external connectivity through a WAN router and for east-west traffic within the fabric push host routes to the fabric that are redistributed within the fabric and towards external interfaces.

Service nodes can peer with the ACI fabric to advertise a Virtual IP (VIP) route that is redistributed and to an external WAN interface.

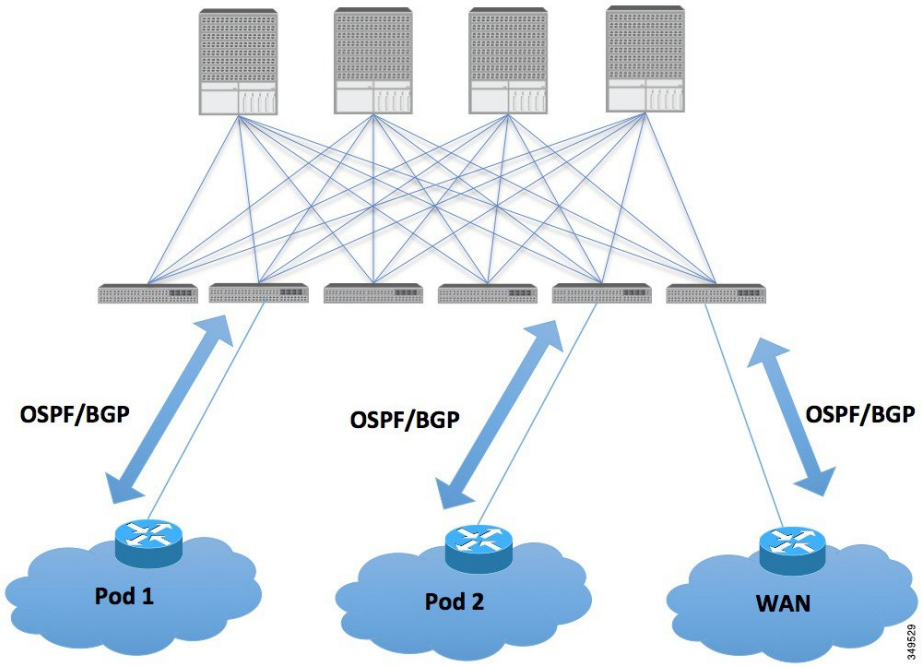
Figure 29: Service Node Transit Connectivity



The VIP is the external facing IP address for a particular site or service. A VIP is tied to one or more servers or nodes behind a service node.

The ACI fabric acts as transit for external connectivity and interconnect between PODs. Cloud providers can deploy managed resource PODs inside a customer data center. The demarcation point can be an L3Out with OSPF/BGP peering with the fabric.

Figure 30: Multi-pod Transit Connectivity



In such scenarios, the policies are administered at the demarcation points and ACI policies need not be imposed.

L4-L7 route peering is a special use case of the fabric as a transit where the ACI fabric serves as a transit OSPF/BGP domain for other PODs. Route Peering is used to configure OSPF/BGP peering on the L4-L7 service device so that it can exchange routes with the ACI leaf node to which it is connected. A common use case for route peering is Route Health Injection where the SLB VIP is advertised over OSPF/iBGP to clients outside the ACI fabric. See Appendix H for a configuration walk-through of this scenario.

Transit Routing Overview

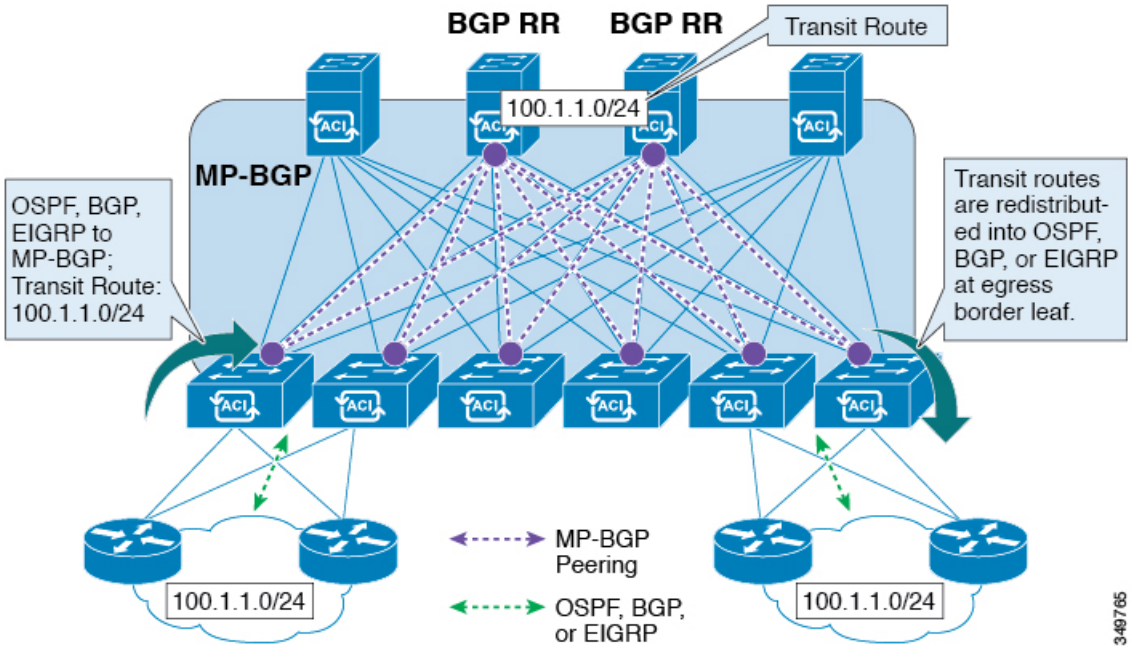
This article provides an overview of Layer 3 transit routing with the Cisco APIC.

The ACI software supports external Layer 3 connectivity with OSPF (NSSA) and iBGP. The ACI fabric advertises the tenant bridge domain subnets out to the external routers on the External Layer 3 Outside connections. The routes that are learned from the external routers are not advertised to the other external routers. The ACI fabric behaves like a stub network and it can be used to carry the traffic between the external Layer 3 domains.

The ACI software adds support for transit routing. Multiple External Layer 3 Outside connections within a single tenant/context (VRF) are supported and the ACI fabric can advertise the routes that are learned from one External Layer 3 Outside connection to another External Layer 3 Outside connection. The external Layer 3 domains peer with the ACI fabric on the leaf switches (border leaves). The fabric is a transit Multiprotocol-Border Gateway Protocol (MP-BGP) domain between the peers.

The ACI fabric configuration for external Layer 3 Outside connections is done at the tenant/VRF level. The routes that are learned from the external peers are imported into MP-BGP at the ingress leaf per VRF. The prefixes that are learned from the External Layer 3 Outside connections are exported to the leaf switches only where the tenant VRF is present.

Figure 31: Transit Routing Overview Diagram



Route Distribution Within the ACI Fabric

ACI supports the following routing mechanisms:

- Static Routes
- OSPFv2 (IPv4)
- OSPFv3 (IPv6)
- iBGP
- eBGP (IPv4 and IPv6)
- EIGRP (IPv4) protocols

ACI supports the VRF-lite implementation when connecting to the external routers. Using sub-interfaces, the border leaf can provide Layer 3 outside connections for the multiple tenants with one physical interface. The VRF-lite implementation requires one protocol session per tenant.

Within the ACI fabric, Multiprotocol BGP (MP-BGP) is implemented between the leaf and the spine switches to propagate the external routes within the ACI fabric. The BGP route reflector technology is deployed in order to support a large number of leaf switches within a single fabric. All of the leaf and spine switches are in one single BGP Autonomous System (AS). Once the border leaf learns the external routes, it can then redistribute the external routes of a given VRF to an MP-BGP address family VPN version 4 or VPN version

6. With address family VPN version 4, MP-BGP maintains a separate BGP routing table for each VRF. Within MP-BGP, the border leaf advertises routes to a spine switch, that is a BGP route reflector. The routes are then propagated to all the leaves where the VRFs (or private network in the APIC GUI's terminology) are instantiated.

External Layer 3 Outside Connection Types

ACI supports the following External Layer 3 Outside connection options:

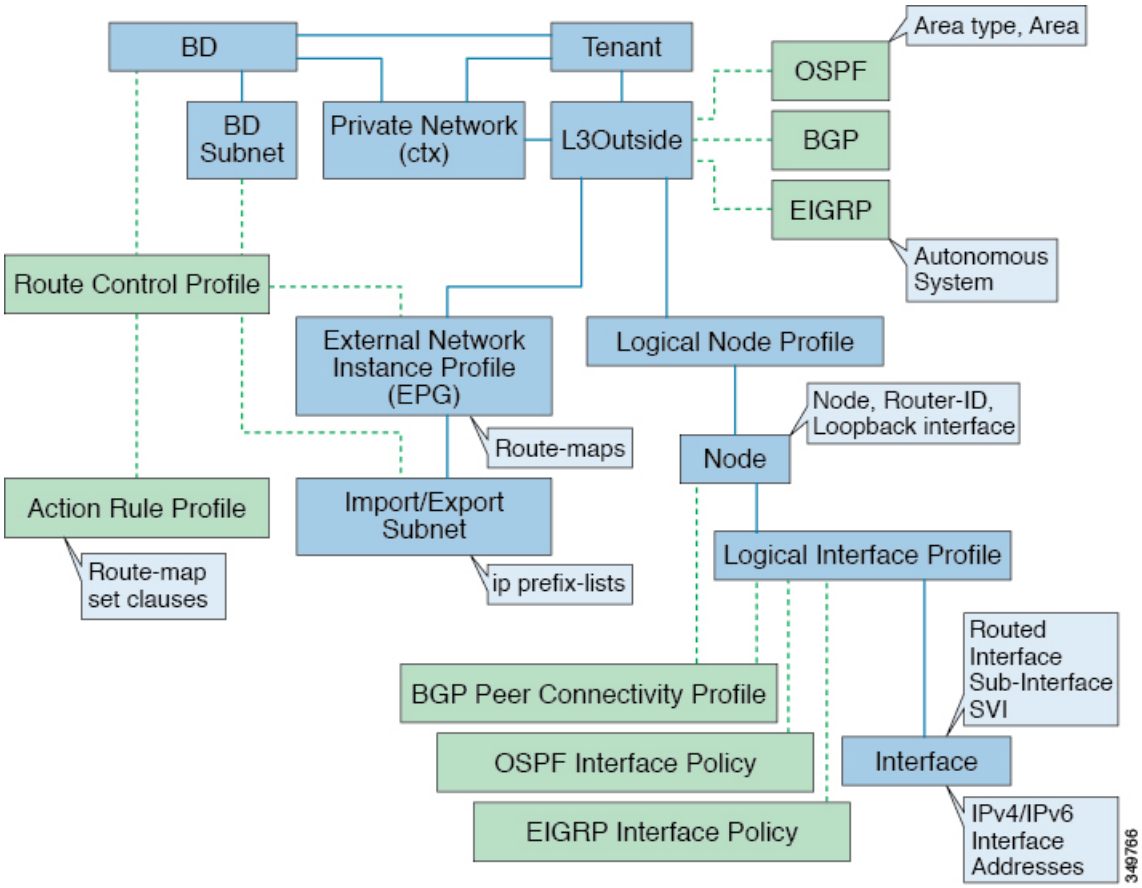
- Static Routing (supported for IPv4 and IPv6)
- OSPFv2 for normal and NSSA areas (IPv4)
- OSPFv3 for normal and NSSA areas (IPv6)
- iBGP (IPv4 and IPv6)
- eBGP (IPv4 and IPv6)
- EIGRP (IPv4 only)

The External Layer 3 Outside connections are supported on the following interfaces:

- Layer 3 Routed Interface
- Sub-interface with 802.1Q tagging - With sub-interface, the same physical interface can be used to provide a Layer 2 outside connection for multiple private networks.

- Switched Virtual Interface (SVI) - With an SVI interface, the same physical interface that supports Layer 2 and Layer 3 and the same physical interface can be used for a Layer 2 outside connection and a Layer 3 outside connection.

Figure 32: ACI Layer 3 Managed Objects



The managed objects that are used for the L3Outside connections are:

- External Layer 3 Outside (L3ext): Routing protocol options (OSPF area type, area, EIGRP AS, BGP), private network, External Physical domain.
- Logical Node Profile: Profile where one or more nodes are defined for the External Layer 3 Outside connections. The router-IDs and the loopback interface configuration is defined in the profile.

Note Use same router-ID for the same node across multiple External Layer 3 Outside connections.

- Logical Interface Profile: IP interface configuration for IPv4 and IPv6 interfaces. It is supported on the Route Interfaces, Routed Sub-Interfaces, and SVIs. The SVIs can be configured on physical ports, port-channels or VPCs.
- OSPF Interface Policy: OSPF Network Type, priority etc.

- EIGRP Interface Policy: Timers, split horizon setting etc
- BGP Peer Connectivity Profile: The profile where most BGP peer settings, remote-as, local-as, and BGP peer connection options are configured. The BGP peer connectivity profile can be associated with the logical interface profile or the loopback interface under the node profile. This determines the update-source configuration for the BGP peering session.
- External Network Instance Profile (EPG) (l3extInstP): The external EPG is also referred to as the prefix based EPG or InstP. The import and export route control policies, security import polices, and contract associations are defined in this profile. Multiple external EPGs can be configured under a single L3Out. Multiple external EPGs may be used when a different route or a security policy is defined on a single External Layer 3 Outside connections. An external EPG or multiple external EPGs combine into a route-map. The import/export subnets defined under the external EPG associate to the IP prefix-list match clauses in the route-map. The external EPG is also where the import security subnets and contracts are associated. This is used to permit or drop traffic for this L3out.
- Action Rules Profile: The action rules profile is used to define the route-map set clauses for the L3Out. The supported set clauses are the BGP communities (standard and extended), Tags, Preference, Metric, and Metric type.
- Route Control Profile: The route-control profile is used to reference the action rules profile(s). This can be an ordered list of action rules profiles. The Route Control Profile can be referenced by a tenant BD, BD subnet, external EPG, or external EPG subnet.

There are additional protocol settings for BGP, OSPF, and EIGRP L3Outs. These settings are configured per tenant in the ACI Protocol Policies section in the GUI.

Supported Transit Combination Matrix

Layer 3 Outside Connection Type		OSPF	iBGP			eBGP		EIGRP	Static Route
			iBGP over OSPF	iBGP over Static route	iBGP over direct connection	eBGP over OSPF	eBGP over direct connection		
OSPF		Yes	Yes*	Yes	X	Yes	Yes	Yes	Yes
iBGP	iBGP over OSPF	Yes*	X	X	X	X	Yes	X	Yes
	iBGP over Static route	Yes	X	X	X	X	**Yes**	X	Yes
	iBGP over direct connection	Yes	X	X	X	X	**Yes**	X	Yes

Layer 3 Outside Connection Type		OSPF	iBGP			eBGP		EIGRP	Static Route
			iBGP over OSPF	iBGP over Static route	iBGP over direct connection	eBGP over OSPF	eBGP over direct connection		
eBGP	eBGP over OSPF	Yes	X	X	**Yes**	Yes	X	X	X
	eBGP over direct connection	Yes	Yes	X	**Yes**	X	Yes	X	Yes
EIGRP		Yes	X	X	X	X	X	X	
Static route		Yes	Yes	Yes	Yes	X	Yes		Yes

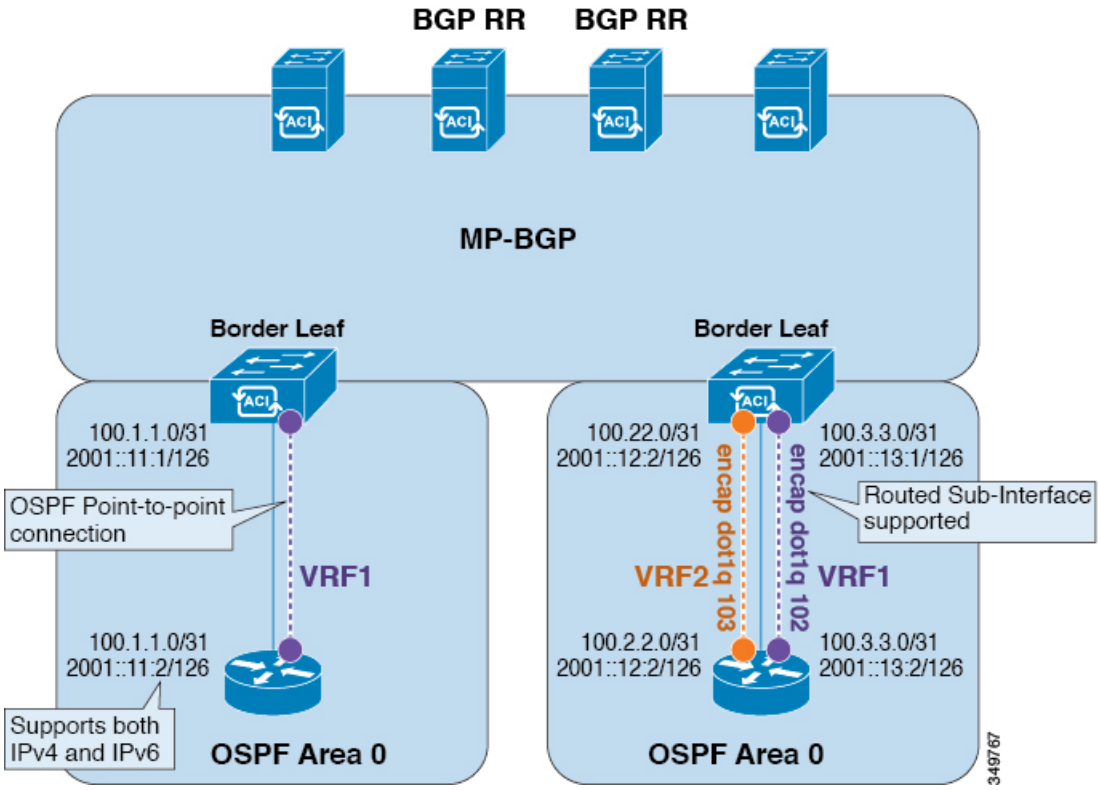
- * = Not supported on the same leaf switch
- X = Unsupported/Untested combinations
- **bold text = Supported in this release**

OSPF Layer 3 Outside Connections

OSPF Layer 3 Outside connections can be normal or NSSA areas. The backbone (area 0) area is also supported as an OSPF Layer 3 Outside connection area. ACI supports both OSPFv2 for IPv4 and OSPFv3 for IPv6. When creating an OSPF Layer 3 Outside, it is not necessary to configure the OSPF version. The correct OSPF process is created automatically based on the interface profile configuration (IPv4 or IPv6 addressing). Both IPv4 and IPv6 protocols are supported on the same interface (dual stack) but it is necessary to create two separate interface profiles.

Layer 3 Outside connections are supported for the routed interfaces, routed sub-interfaces, and SVIs. The SVIs are used when there is a need to share the physical connect for both L2 and L3 traffic. The SVIs are supported on ports, port-channels, and VPC port-channels.

Figure 33: OSPF Layer3 Out Connections



When an SVI is used for an Layer 3 Outside connection, an external bridge domain is created on the border leaf switches. The external bridge domain allows connectivity between the two VPC switches across the ACI fabric. This allows both the VPC switches to establish the OSPF adjacencies with each other and the external OSPF device.

When running OSPF over a broadcast network, the time to detect a failed neighbor is the dead time interval (default 40 seconds). Reestablishing the neighbor adjacencies after a failure may also take longer due to designated router (DR) election.

Note A link or port-channel failure to one VPC Node does not cause an OSPF adjacency to go down. The OSPF adjacency can stay up via the external BD accessible through the other VPC node.

EIGRP Layer 3 Outside Connections

EIGRP Layer 3 Outside connections are supported on the same interface types as OSPF except that IPv6 is not supported for EIGRP.

Note VPC/SVI configuration for EIGRP is the same as OSPF.

BGP Protocol Peering to External BGP Speakers

ACI supports peering between the border leaves and the external BGP speakers using iBGP and eBGP. ACI supports the following connections for BGP peering:

- iBGP peering over OSPF
- eBGP peering over OSPF
- iBGP peering over direct connection
- eBGP peering over direct connection
- iBGP peering over static route

Note When OSPF is used with BGP peering, OSPF is only used to learn and advertise the routes to the BGP peering addresses. All route control applied to the Layer 3 Outside Network (EPG) are applied at the BGP protocol level.

ACI supports a number of features for iBGP and eBGP connectivity to external peers. The BGP features are configured on the **BGP Peer Connectivity Profile**.

The BGP peer connectivity profile features are described in the following table:

Table 3: BGP Peer Connectivity Profile Features

BGP Features	Feature Description	NX-OS Equivalent Commands
Allow Self-AS	Works with Allowed AS Number Count setting.	**allowas-in**
Disable peer AS check	Disable checking of the peer AS number when advertising.	**disable-peer-as-check**
Next-hop self	Always set the next hop attribute to the local peering address.	**next-hop-self**
Send community	Send the community attribute to the neighbor.	**send-community**
Send community extended	Send the extended community attribute to the neighbor.	**send-community extended**
Password	The BGP MD5 authentication.	**password**

BGP Features	Feature Description	NX-OS Equivalent Commands
Allowed AS Number Count	Works with Allow Self-AS feature.	**allowas-in**
Disable connected check	Disable connected check for the directly connected EBGP neighbors (allowing EBGP neighbor peering from the loopbacks).	
TTL	Set the TTL value for EBGP multihop connections. It is only valid for EBGP.	**ebgp-multihop <TTL>**
Autonomous System Number	Remote Autonomous System number of the peer.	**neighbor <x.x.x.x> remote-as**
Local Autonomous System Number Configuration	Options when using the Local AS feature. (No Prepend+replace-AS+dual-AS etc).	
Local Autonomous System Number	The local AS feature used to advertise a different AS number than the AS assigned to the fabric MP-BGP Route Reflector Profile. It is only supported for the EBGP neighbors and the local AS number must be different than the route reflector policy AS.	**local-as xxx <no-prepend> <replace-as> <dual-as>**

Transit Route Control

An ACI fabric can have multiple external Layer 3 connections per tenant/VRF running dynamic routing protocols (OSPF, EIGRP, and BGP). Route control policies are implemented in the ACI fabric to control the distribution of the routes that are learned from an External Layer 3 Outside connection or configured as a static route. ACI supports import and export route control. Import and export route control uses the route-maps and the IP prefix-lists to control the import and export of the prefixes that are allowed into and advertised out of the ACI fabric.

The default setting for import route control is to allow all the prefixes. All leaf switches in the ACI fabric learn of all the external prefixes where that VRF is deployed. The default setting for the export route control is to deny all the prefixes. The import route control can be enabled but is only supported for BGP. All OSPF and EIGRP learned routes are allowed into their respective protocol on the border leaf where the Layer 3 Outside connection is deployed. These prefixes are redistributed (imported) into MP-BGP at the ingress border leaf per tenant/VRF.

Import Route Control

- Controls the import of prefixes into the routing table on the ingress leaf.
- Disabled by default.
- Only supported for BGP.
- Implemented with an input route-map associated to the external BGP neighbor.

Export Route Control

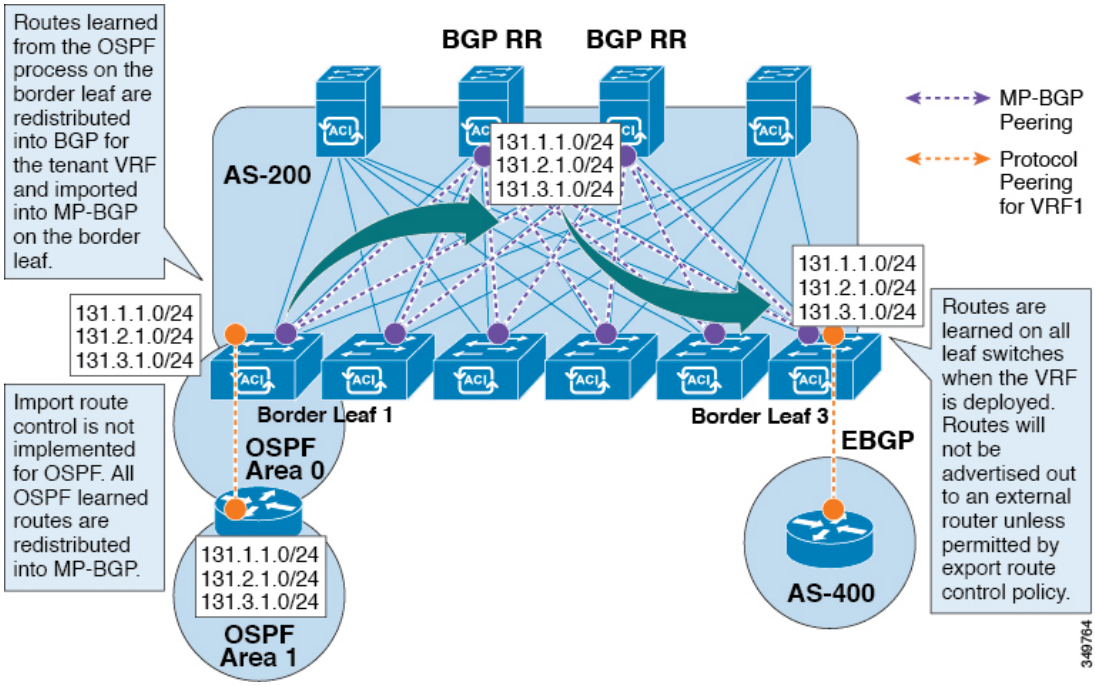
- Controls the export of transit prefixes advertised out of the ACI fabric (over Layer 3 Outside connections).
- Supported for all Layer 3 Outside connection types.
- Always enabled.
- Default setting is to deny all the prefixes.
- Implemented with redistribution route-map (OSPF/EIGRP) and neighbor route-map (BGP).
- Not used to control the export of the tenant subnets or the originating default route.

Import and export route control is configured under the External Network Instance Profile (l3extInstP).

Note The import/export route control is used to control the import and export of the transit route prefixes (the routes that are learned from the external Layer 3 devices) and the static routes. The import/export route control is not used for the tenant subnets (the subnets that are configured under the tenant bridge domains) or when originating a default route.

ACI Route Redistribution

Figure 34: ACI Route Redistribution



- The routes that are learned from the OSPF process on the border leaf are redistributed into BGP for the tenant VRF and they are imported into MP-BGP on the border leaf.
- The import route control is not implemented for OSPF. All OSPF learned routes are redistributed into MP-BGP.
- The routes are learned on the border leaf where the VRF is deployed. The routes are not advertised to the External Layer 3 Outside connection unless it is permitted by the export route control.

Note When a subnet for a bridge domain/EPG is set to Advertise Externally, the subnet is programmed as a static route on a border leaf. When the static route is advertised, it is redistributed into the EPG's Layer 3 outside network routing protocol as an external network, not injected directly into the routing protocol.

Controls Enabled for Subnets Configured under the Layer 3 Outside Network Instance Profile

The following controls can be enabled for the subnets that are configured under the Layer 3 Outside Network Instance Profile.

Table 4: Route Control Options

Route control Setting	Use	Options
Export Route Control	To allow the prefixes that are advertised to the external peers. Implemented with IP prefix-lists.	Specific match (prefix and prefix length).
Import Route Control	To allow prefixes that are inbound from the external BGP peers. Implemented with IP prefix-lists.	Specific match (prefix and prefix length) .
Security Import Subnet	To permit the packets between two prefix based EPGs. Implemented with ACLs.	Uses the ACL match prefix/wildcard match rules.
Aggregate Export	To allow all prefixes to be advertised to the external peers. Implemented with 0.0.0.0/ le 32 IP prefix-list.	Only supported for 0.0.0.0/0 subnet (all prefixes).
Aggregate Import	To allow all prefixes that are inbound from an external BGP peer. Implemented with 0.0.0.0/0 le 32 IP prefix-list.	Only supported for 0.0.0.0/0 subnet (all prefixes).

In many cases, it may be preferred to advertise all the transit routes out to an Layer 3 Outside connection. In this case, the aggregate export option is used with the prefix 0.0.0.0/0. This creates an IP prefix-list entry (permit 0.0.0.0/0 le 30) that is configured as a match clause in the export route-map. Use **show route-map <outbound route-map>** command and **show ip prefix-list <match-clause>** to view the output.

Advertising Tenant BD Subnets Outside the Fabric

The import and export route control policies only apply to the transit routes (the routes that are learned from other external peers) and the static routes. The subnets internal to the fabric that are configured on the tenant BD subnets are not advertised out using the export policy subnets. The tenant subnets are still permitted using the IP prefix-lists and the route-maps but they are implemented using different configuration steps. See the following configuration steps to advertise the tenant subnets outside the fabric:

Procedure

Step 1 Configure the tenant subnet scope as **Public Subnet** in the subnet properties window.

Step 2 (Optional) Set the Subnet Control as **ND RA Prefix** in the subnet properties window.

Step 3 Associate the tenant bridge domain (BD) with external Layer 3 Outside.

Step 4 Create contract (provider/consumer) association between the tenant EPG and the external EPG.
Setting the BD subnet to scope Public and associating the BD to the Layer 3 Outside creates an IP prefix-list and the route-map sequence entry on the border leaf for the BD subnet prefix.

Tenant EPG to Layer 3 Outside Contract

The tenant EPG needs a contract provider/consumer association with the Layer 3 Outside connection. It creates a route entry for the subnet on the border leaf (If the tenant BD is not previously deployed on the leaf) and it is also used to permit the traffic in the data plane.

In some cases, the tenant subnet may be advertised out to the external peer even if no contract is configured. The tenant subnet is advertised out if any of the following conditions are true:

- The tenant EPG/BD is already deployed on the border leaf.
- OR the tenant EPG/BD has a contract with a tenant/EPG deployed on the border leaf.

These two conditions create an entry in the routing table for the tenant subnet and the Public scope and the Layer 3 Outside association allows the subnet to be advertised out but the data plane traffic is not permitted without a contract.

Note This entry is valid only if the tenant private network (context) is set with Policy Control Enforcement set to enforced. If Policy Control Enforcement is set to unenforced, the tenant prefixes are present on the border leaf without any contracts.

Advertising a Default Route

For external connections to the fabric that only require a default route, there is support for originating a default route for OSPF, EIGRP, and BGP Layer 3 Outside connections. If a default route is received from an external peer, this route can be redistributed out to another peer following the transit export route control as described earlier in this article.

A default route can also be advertised out using a Default Route Leak Policy. This policy supports advertising a default route if it is present in the routing table or it supports advertising a default route always. The Default Route Leak Policy is configured at the Layer 3 Outside connection.

When creating a Default Route Leak Policy, follow these guidelines:

- For BGP, the Always property is not applicable.
- For BGP, when choosing the Scope property, you must choose Outside.
- For OSPF, the Scope value Context creates a type-5 LSA while the Scope value Outside creates type-7 LSA. This selection depends on the area type being used in that Layer3 outside. Therefore, if the area type is regular, set the scope to Context and if the area type is NSSA, set the scope to Outside.

Route Control Profile Policies

The ACI fabric also supports the route-map set clauses for the routes that are advertised into and out of the fabric. The route-map set rules are configured with the Route Control Profile policies and the Action Rule Profiles.

ACI supports the following set options:

Table 5: Action Rule Profile Properties (route-map set clauses)

Property	OSPF	EIGRP	BGP	Comments
Set Community			Yes	Supports regular and extended communities.
Route Tag	Yes	Yes		Supported only for BD subnets. Transit prefixes are always assigned the tag 4294967295.
Preference			Yes	Sets BGP local preference.
Metric	Yes		Yes	Sets MED for BGP. Will change the metric for EIGRP but you cannot specify the EIGRP composite metric.
Metric Type	Yes			OSPF Type-1 and OSPF Type-2.

The Route Profile Polices are created under the Layer 3 Outside connection. A Route Control Policy can be referenced by the following objects:

- Tenant BD Subnet
- Tenant BD
- External EPG
- Externel EPG import/export subnet

Here is an example of using Import Route Control for BGP and setting the local preference for an external route learned from two different Layer 3 Outsides. The Layer 3 Outside connection for the external connection to AS300 is configured with the Import Route Control enforcement. An action rule profile is configured to set the local preference to 200 in the Action Rule Profile for Local Preference window.

The Layer 3 Outside connection External EPG is configured with a 0.0.0.0/0 import aggregate policy to allow all the routes. This is necessary because the import route control is enforced but any prefixes should not be blocked. The import route control is enforced to allow setting the local preference. Another import subnet 151.0.1.0/24 is added with a Route Profile that references the Action Rule Profile in the External EPG settings for Route Control Profile window.

Use the **show ip bgp vrf overlay-1** command to display the MP-BGP table. The MP-BGP table on the spine displays the prefix 151.0.1.0/24 with local preference 200 and a next hop of the border leaf for the BGP 300 Layer 3 Outside connection.

There are two special route control profiles—default-import and default-export. If the user configures using the names default-import and default-export, then the route control profile is automatically applied at the Layer3 outside level for both import and export. The default-import and default-export route control profiles cannot be configured using the 0.0.0.0/0 aggregate.

A route control profile is applied in the following sequential order for fabric routes:

1. Tenant BD subnet
2. Tenant BD
3. Layer3 outside

The route control profile is applied in the following sequential order for transit routes:

1. External EPG prefix
2. External EPG
3. Layer3 outside

Note When you configure Layer 3 Outside (L3Out) connections to external routers, it is critical that the MTU be set appropriately on both sides. On some platforms, such as ACI, Cisco NX-OS, and Cisco IOS, the configurable MTU value takes into account packet headers (resulting in a max packet size to be set as 9000 bytes), whereas other platforms such as IOS-XR configure the MTU value exclusive of packet headers (resulting in a max packet size of 8986 bytes). For the appropriate MTU values for each platform, see the relevant configuration guides. Cisco highly recommends you test the MTU using CLI-based commands. For example, on the Cisco NX-OS CLI, use a command such as `ping 1.1.1.1 df-bit packet-size 9000 source-interface ethernet 1/1`

Security Import Policies

The policies discussed in the documentation have dealt with the exchange of the routing information into and out of the ACI fabric and the methods that are used to control and tag the routes. The ACI fabric operates in a whitelist model where the default behavior is to drop all the data plane traffic between the endpoint groups unless explicitly permitted by the policy. This whitelist model applies to the external EPGs and the tenant EPGs.

There are some differences in how the security policies are configured and how they are implemented for the transit traffic compared to the tenant traffic:

Transit Security Policies

- Uses prefix filtering.
- Does not support Ethertype, protocol, L4 port, and TCP flag filters.
- Implemented with the security import subnets (prefixes) and the contracts that are configured under the external EPG.

Tenant EPG Security Policies

- Does not use prefix filtering.
- Supports Ethertype, protocol, L4 port, and TCP flag filters.
- Supported for tenant EPG←→EPG and tenant EPG←→External EPGs.

If there are no contracts between the external prefix based EPGs, the traffic is dropped. Allowing traffic between the two external EPGs requires configuring a contract and a security prefix. As only prefix filtering is supported, the default filter can be used.

External Layer 3 Outside Connection Contracts

The union of prefixes for Layer 3 Outside connections are programmed on all the leaf nodes where the Layer 3 Outside connections are deployed. When more than two Layer 3 Outside connections are deployed, the use of the catch all rule 0.0.0.0/0 can allow traffic between the Layer 3 Outside connections that do not have a contract.

Configuring the Provider/Consumer contract associations and the security import subnets is done at the External Layer 3 Outside connection Instance Profile (instP).

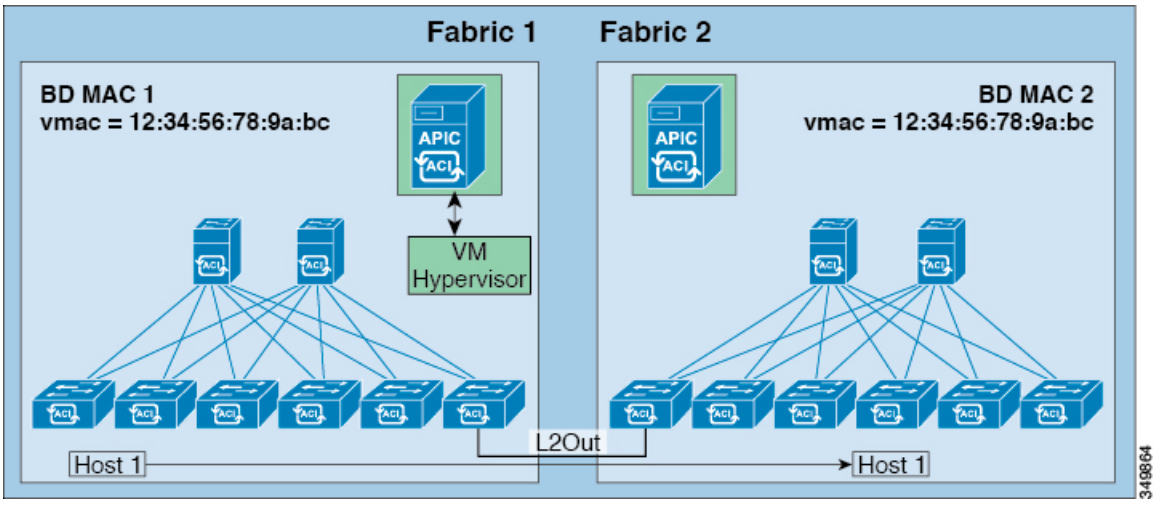
When security import subnets are configured and the catch all rule 0.0.0.0/0 is supported, the security import subnets follow the ACL type rules. The security import subnet rule 10.0.0.0/8 matches all the addresses from 10.0.0.0 – 10.255.255.255. It is not required to configure an exact prefix match for the prefixes that are permitted by the route control subnets.

Care should be taken when configuring the security import subnets if more than two Layer 3 Outside connections are configured in the same VRF, due to the union of the rules.

Common Pervasive Gateway

Multiple ACI fabrics can be configured with an IPv4 common gateway on a per bridge domain basis. Doing so enables moving one or more virtual machines (VM) or conventional hosts across the fabrics while the host retains its IP address. VM host moves across fabrics can be done automatically by the VM hypervisor. The ACI fabrics can be co-located, or provisioned across multiple sites. The Layer 2 connection between the ACI fabrics can be a local link, or can be across a routed WAN link. The following figure illustrates the basic common pervasive gateway topology.

Figure 35: ACI Multi-Fabric Common Pervasive Gateway



The per-bridge domain common pervasive gateway configuration requirements are as follows:

- The bridge domain MAC (*mac*) values for each fabric must be unique.

Note The default bridge domain MAC (*mac*) address values are the same for all ACI fabrics. The common pervasive gateway requires an administrator to configure the bridge domain MAC (*mac*) values to be unique for each ACI fabric.

- The bridge domain virtual MAC (*vmac*) address and the subnet virtual IP address must be the same across all ACI fabrics for that bridge domain. Multiple bridge domains can be configured to communicate across connected ACI fabrics. The virtual MAC address and the virtual IP address can be shared across bridge domains.

Configuring Common Pervasive Gateway Using the GUI

Before You Begin

- The tenant, and VRF are created.
- The bridge domain virtual MAC address and the subnet virtual IP address must be the same across all ACI fabrics for that bridge domain. Multiple bridge domains can be configured to communicate across connected ACI fabrics. The virtual MAC address and the virtual IP address can be shared across bridge domains.
- The Bridge domain that is configured to communicate across ACI fabrics must be in **flood** mode
- Only one EPG from a bridge domain (If the BD has multiple EPGs) should be configured on a border Leaf on the port which is connected to the second Fabric.
- Do not connect hosts directly to an inter-connected Layer 2 network that enables a pervasive common gateway among the two ACI fabrics.

Procedure

Step 1 On the menu bar, click **TENANTS**.

Step 2 In the **Navigation** pane, expand the ***Tenant_name*** > **Networking** > **Bridge Domains**.

Step 3 Right-click **Bridge Domains**, and click **Create Bridge Domain**.

Step 4 In the **Create Bridge Domain** dialog box, perform the following actions and select the appropriate attributes:

a) In the **Name** field, enter a name for the bridge domain.

b) Expand **Subnets**, and in the **Create Subnets** dialog box, in the **Gateway IP** field, enter the IP address. In the **Treat as virtual IP address** field, check the check box. Click **Ok** and click **Finish**.

c) Expand **Subnets** again, and in the **Create Subnets** dialog box, to create the Physical IP address in the **Gateway IP** field, using the same subnet which is configured as the Virtual IP address.

Note The Physical IP address must be unique across ACI fabric

Step 5 Double click on the **Bridge Domain** that you just created in the **Work** pane, and perform the following actions:

a) Click on the **Virtual MAC Address** field, and change **not-applicable** to the appropriate value. Click **Submit**.

Note The default BD MAC address values are the same for all ACI fabrics; this configuration requires the bridge domain MAC values to be unique for each ACI fabric.

Confirm that the bridge domain MAC (pmac) values for each fabric are unique.

Step 6 Create a L2out EPG to extend the BD to other Fabric by right clicking on **External Bridged Networks** and open the **Create Bridged Outside** dialog box, and perform the following actions:

a) In the **Name** field, enter a name for the bridged outside.
b) In the **Bridge Domain** field, select the bridge domain already previously created.
c) In the **Encap** field, enter the VLAN encapsulation to match the other fabric l2out encapsulation.
d) In the **Path Type** field, select **Port**, **PC**, or **VPC** to deploy the EPG and click **Next**.
e) To create an External EPG network click in the **Name** field, enter a name for the network and you can specify the QoS class and click **Finish** to complete Common Pervasive configuration.

Configuring Common Pervasive Gateway Using the REST API

Before You Begin

- The tenant, VRF, and bridge domain are created.

Procedure

Configure Common Pervasive Gateway.

Example:

```
<!-Things that are bolded only matters-->
<?xml version="1.0" encoding="UTF-8"?>
<!-- api/policymgr/mo/.xml -->
<polUni>
  <fvTenant name="test">
    <fvCtx name="test"/>

    <fvBD name="test" vmac="12:34:56:78:9a:bc">
      <fvRsCtx tnFvCtxName="test"/>
      <!-- Primary address -->
      <fvSubnet ip="192.168.15.254/24" preferred="yes"/>
      <!-- Virtual address -->
      <fvSubnet ip="192.168.15.1/24" virtual="yes"/>
    </fvBD>

    <fvAp name="test">
      <fvAEPg name="web">
        <fvRsBd tnFvBDName="test"/>
        <fvRsPathAtt tDn="topology/pod-1/paths-101/pathep-[eth1/3]" encap="vlan-1002"/>
      </fvAEPg>
    </fvAp>
  </fvTenant>
</polUni>
```

Configuring Common Pervasive Gateway Using the NX-OS Style CLI

Before You Begin

- The tenant, VRF, and bridge domain are created.

Procedure

Configure Common Pervasive Gateway.

Example:

```
apic1#configure
apic1(config)#tenant demo
apic1(config-tenant)#bridge-domain test
apic1(config-tenant-bd)#l2-unknown-unicast flood
apic1(config-tenant-bd)#arp flooding
apic1(config-tenant-bd)#exit

apic1(config-tenant)#interface bridge-domain test
apic1(config-tenant-interface)#multi-site-mac-address 12:34:56:78:9a:bc
apic1(config-tenant-interface)#mac-address 00:CC:CC:CC:C1:01 (Should be unique for each ACI
 fabric)
apic1(config-tenant-interface)#ip address 192.168.10.1/24 multi-site
apic1(config-tenant-interface)#ip address 192.168.10.254/24 (Should be unique for each ACI
 fabric)
```

INDEX

F

I

L

M

R

S

T

V