



Access Interfaces

This chapter contains the following sections:

- [Physical Ports, on page 1](#)
- [Port Channels, on page 5](#)
- [Virtual Port Channels, on page 14](#)
- [FEX Interfaces, on page 25](#)

Physical Ports

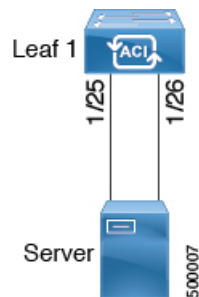
Configuring Leaf Switch Physical Ports Using the Advanced GUI

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 1: 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 Device Type* domain is required for enabling an EPG to use the interfaces specified in the switch profile.

- Specify *individual* as the interface type to use.
- Specify the interface ID to use.
- Specify the interface policies to use.
- Specify the attached device type to use. Choose **Bare Metal** for connecting bare metal servers. Bare metal uses the **phys** domain type.
- 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.

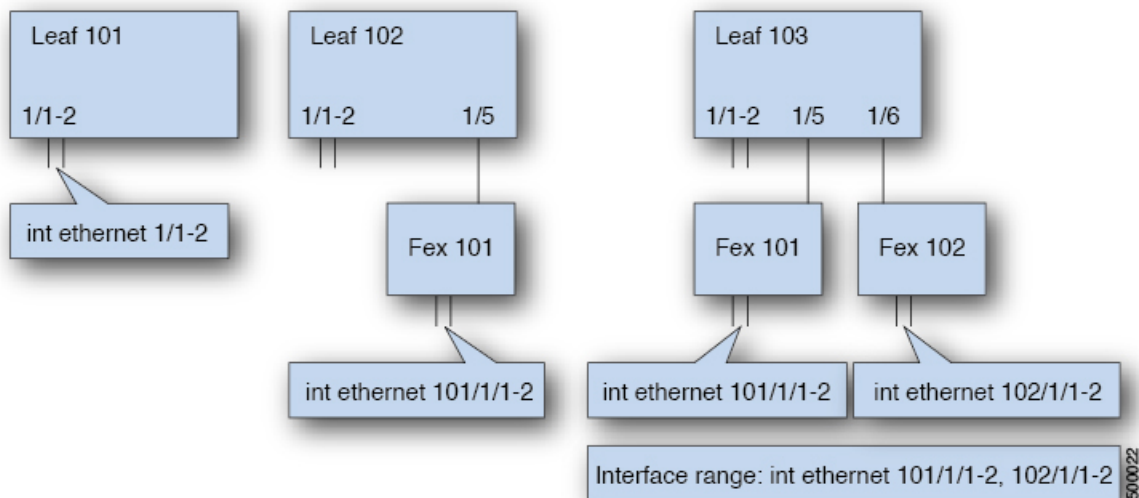


Note While this configuration enables hardware connectivity, no data traffic can flow without a valid application profile, EPG, and contract that is associated with this hardware configuration.

Configuring Physical Ports in Leaf Nodes

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: apicl# configure	Enters global configuration mode.
Step 2	leaf node-id Example: apicl(config)# leaf 102	Specifies the leaf or leaves to be configured. The <i>node-id</i> can be a single node ID or a range of IDs, in the form <i>node-id1 - node-id2</i> , to which the configuration will be applied.
Step 3	interface type Example: apicl(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	(Optional) fex associate node-id Example: apicl(config-leaf-if)# fex associate 101	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.

	Command or Action	Purpose
Step 5	speed <i>speed</i> 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 <i>speedValue</i>	Set the speed for physical interface
[no] link debounce time <i>time</i>	Set link debounce
[no] negotiate auto	Configure negotiate
[no] cdp enable	Disable/enable Cisco Discovery Protocol (CDP) Note CDP is not supported with FEX switches (in this case, use LLDP).
[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
spanning-tree {bpduguard bpdupfilter} {enable disable}	Configure spanning tree BPDU
[no] storm-control level <i>percentage</i> [burst-rate <i>percentage</i>]	Storm-control configuration (percentage)
[no] storm-control pps <i>packets-per-second</i> burst-rate <i>packets-per-second</i>	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
```

Port Channels

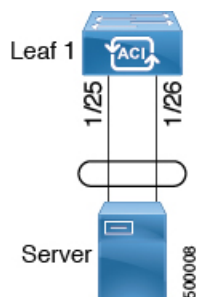
ACI Leaf Switch Port Channel Configuration Using the Advanced GUI

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 2: 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, 1 Gbit port speed), Storm Control Interface Policy, and so forth.

Note The *Attached Device Type* is required for enabling an EPG to use the interfaces specified in the switch profile.

- Specify *pc* as the interface type to use.
- Specify the interface IDs to use.
- Specify the interface policies to use.
- Specify the attached device type to use. Choose **Bare Metal** for connecting bare metal servers. Bare metal uses the **phys** domain type.
- 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.

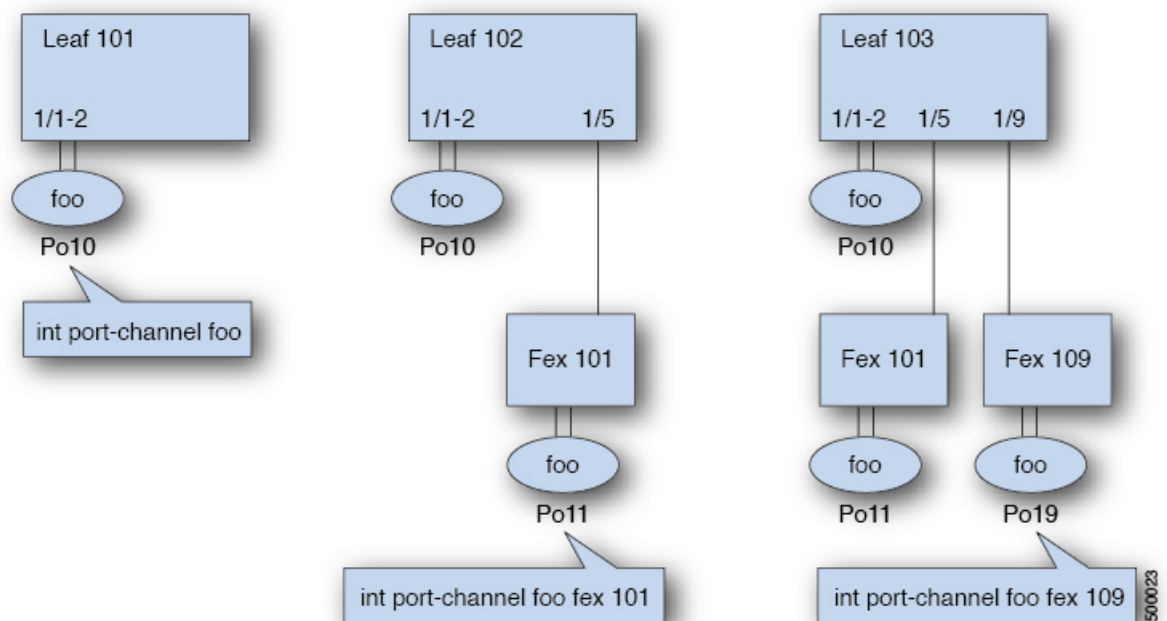


Note While this configuration enables hardware connectivity, no data traffic can flow without a valid application profile, EPG, and contract that is associated with this hardware configuration.

Configuring Port Channels in Leaf Nodes

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 <i>channel-name</i> Example: apic1(config)# template port-channel foo	Creates a new port-channel or configures an existing port-channel (global configuration).
Step 3	[no] switchport access vlan <i>vlan-id</i> tenant <i>tenant-name</i> application <i>application-name</i> epg <i>epg-name</i> Example: apic1(config-po-ch-if)# switchport access vlan 4 tenant ExampleCorp application Web epg webEpg	Deploys the EPG with the VLAN on all ports with which the port-channel is associated.
Step 4	channel-mode active Example: apic1(config-po-ch-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 5	exit Example: apic1(config-po-ch-if)# exit	Returns to configure mode.
Step 6	leaf <i>node-id</i> Example: apic1(config)# leaf 101	Specifies the leaf switches to be configured. The <i>node-id</i> can be a single node ID or a range of IDs, in the form <i>node-id1</i> – <i>node-id2</i> , to which the configuration will be applied.
Step 7	interface <i>type</i> 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 8	[no] channel-group <i>channel-name</i> 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.

	Command or Action	Purpose
Step 9	(Optional) lacp port-priority <i>priority</i> Example: <pre>apicl(config-leaf-if)# lacp port-priority 1000 apicl(config-leaf-if)# lacp rate fast</pre>	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 Note CDP is not supported with FEX switches (in this case, use LLDP).
[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
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 1
[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

CLI Syntax	Feature
[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).

```
apicl(config)# template port-channel foo
apicl(config-po-ch-if)# switchport access vlan 4 tenant ExampleCorp application Web epg webEpg
apicl(config-po-ch-if)# speed 10G
apicl(config-po-ch-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.

```
apicl(config)# leaf 102
apicl(config-leaf)# interface ethernet 101/1/1-2
apicl(config-leaf-if)# channel-group foo
```

In Leaf 102, this port channel interface can be referred to as interface port-channel foo FEX 101.

```
apicl(config)# leaf 102
apicl(config-leaf)# interface port-channel foo fex 101
apicl(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.

```
apic1(config)# leaf 101-103
apic1(config-leaf)# interface ethernet 1/1-2
apic1(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.

```
apic1(config)# leaf 101-103
apic1(config-leaf)# interface ethernet 1/3-4
apic1(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.

```
apic1(config)# leaf 101-103
apic1(config-leaf)# interface eth 1/2,1/4
apic1(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>.

```
apic1(config)# leaf 101
apic1(config-leaf)# interface ethernet 1/1-2
apic1(config-leaf-if)# channel-group foo
apic1(config-leaf-if)# exit
apic1(config-leaf)# exit
apic1(config)# leaf 102
apic1(config-leaf)# interface ethernet 1/3-4
apic1(config-leaf-if)# channel-group foo
apic1(config-leaf-if)# exit
apic1(config-leaf)# exit
apic1(config)# leaf 103
apic1(config-leaf)# interface ethernet 1/5-8
apic1(config-leaf-if)# channel-group foo
apic1(config-leaf-if)# exit
apic1(config-leaf)# interface ethernet 101/1/1-2
apic1(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.

```
apic1(config)# leaf 101
apic1(config-leaf)# interface ethernet 1/1-2
```

```
apic1(config-leaf-if)# channel-group foo
apic1(config-leaf-if)# lacp port-priority 1000
apic1(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
apic1(config)# leaf 101-103
apic1(config-leaf)# interface ethernet 1/3-4
apic1(config-leaf-if)# channel-group foo

// configure admin state in specific leaf
apic1(config)# leaf 101
apic1(config-leaf)# interface port-channel foo
apic1(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
apic1(config)# interface port-channel foo
apic1(config-if)# speed 1G
apic1(config-if)# channel-mode active

// create port-channel foo in each leaf
apic1(config)# leaf 101-103
apic1(config-leaf)# interface ethernet 1/1-2
apic1(config-leaf-if)# channel-group foo

// override port-channel foo in leaf 102
apic1(config)# leaf 102
apic1(config-leaf)# interface port-channel foo
apic1(config-leaf-if)# speed 10G
apic1(config-leaf-if)# channel-mode on
apic1(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.

```
apic1(config)# leaf 101-103
apic1(config-leaf)# interface ethernet 1/3-4
apic1(config-leaf-if)# channel-group foo
apic1(config-leaf-if)# channel-group bar
```

Configuring Two Port Channels Applied to Multiple Switches Using the REST API

This example creates two port channels (PCs) on leaf switch 17, another port channel on leaf switch 18, and a third one on leaf switch 20. On each leaf switch, the same interfaces will be part of the PC (interface 1/10 to 1/15 for port channel 1 and 1/20 to 1/25 for port channel 2). The policy uses two switch blocks because

each a switch block can contain only one group of consecutive switch IDs. All these PCs will have the same configuration.



Note Even though the PC configurations are the same, this example uses two different interface policy groups. Each Interface Policy Group represents a PC on a switch. All interfaces associated with a given interface policy group are part of the same PCs.

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 and protocol(s) are configured and available.

Procedure

To create the two PCs, send a post with XML such as the following:

Example:

```
<infraInfra dn="uni/infra">
  <infraNodeP name="test">
    <infraLeafS name="leafs" type="range">
      <infraNodeBlk name="nblk"
        from_"17" to_"18"/>
      <infraNodeBlk name="nblk"
        from_"20" to_"20"/>
    </infraLeafS>
    <infraRsAccPortP tDn="uni/infra/accportprof-test1"/>
    <infraRsAccPortP tDn="uni/infra/accportprof-test2"/>
  </infraNodeP>
  <infraAccPortP name="test1">
    <infraHPortS name="pselc" type="range">
      <infraPortBlk name="blk1"
        fromCard="1" toCard="1"
        fromPort="10" toPort="15"/>
    <infraRsAccBaseGrp
      tDn="uni/infra/funcprof/accbundle-bndlgrp1"/>
    </infraHPortS>
  </infraAccPortP>
  <infraAccPortP name="test2">
    <infraHPortS name="pselc" type="range">
      <infraPortBlk name="blk1"
        fromCard="1" toCard="1"
        fromPort="20" toPort="25"/>
    <infraRsAccBaseGrp
      tDn="uni/infra/funcprof/accbundle-bndlgrp2" />
    </infraHPortS>
  </infraAccPortP>
</infraFuncP>
```

```

<infraAccBndlGrp name="bndlgrp1" lagT="link">
  <infraRsHIfPol tnFabricHIfPolName="default"/>
  <infraRsCdpIfPol tnCdpIfPolName="default"/>
  <infraRsLacpPol tnLacpLagPolName="default"/>
</infraAccBndlGrp>

<infraAccBndlGrp name="bndlgrp2" lagT="link">
  <infraRsHIfPol tnFabricHIfPolName="default"/>
  <infraRsCdpIfPol tnCdpIfPolName="default"/>
  <infraRsLacpPol tnLacpLagPolName="default"/>
</infraAccBndlGrp>
</infraFuncP>
</infraInfra>

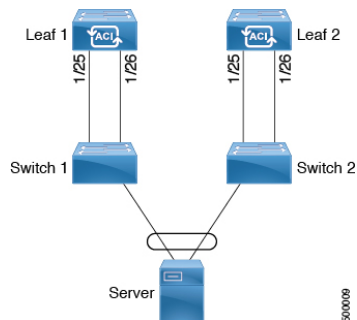
```

Virtual Port Channels

ACI Virtual Port Channel Workflow

This workflow provides an overview of the steps required to configure a virtual port channel (VPC).

Figure 3: 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.



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.

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 the interfaces the virtual port channel will use. Add the Interface Policy parameters, such as group port speed, storm control, CDP, LLDP. Add 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.

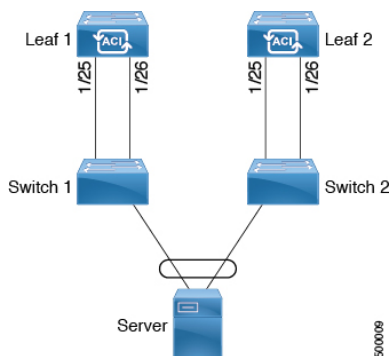
ACI Leaf Switch Virtual Port Channel Configuration Using the Advanced GUI

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 4: Switch Virtual Port Channel Configuration



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 Device 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

A Virtual Port Channel (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.



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.

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 'corp' 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 corp.
- 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 Reg fex 101 101
```

or

```
(config)# vpc context leaf 101 102 fex 101 101
(config-vpc)# interface vpc Reg
```

In the ACI model, VPC configuration is done in the following steps (as shown in the examples below).



Note A VLAN domain is required with a VLAN range. It must be associated with the port-channel template.

1. VLAN domain configuration (global config) with VLAN range
2. VPC domain configuration (global config)

3. Port-channel template configuration (global config)
4. Associate the port-channel template with the VLAN domain
5. Port-channel configuration for VPC (global config)
6. Configure ports to VPC in leaf nodes
7. 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	vlan-domain <i>name</i> [dynamic] [type <i>domain-type</i>] Example: apic1(config)# vlan-domain dom1 dynamic	Configures a VLAN domain for the virtual port-channel (here with a port-channel template).
Step 3	vlan range Example: apic1(config-vlan)# vlan 1000-1999 apic1(config-vlan)# exit	Configures a VLAN range for the VLAN domain and exits the configuration mode. The range can be a single VLAN or a range of VLANs.
Step 4	vpc domain explicit <i>domain-id leaf node-id1</i> <i>node-id2</i> 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: <ul style="list-style-type: none"> • 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 <i>domain-start leaf start-node end-node</i> This command configures a VPC domain consecutively for a selected set of leaf node pairs.
Step 5	peer-dead-interval <i>interval</i> Example:	Sets the interval between hello packets from a neighbor before the router declares the neighbor as down. The range of valid values

	Command or Action	Purpose
	<code>apicl(config-vpc)# peer-dead-interval 10</code>	is 5 to 600 seconds. The 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 6	exit Example: <code>apicl(config-vpc)# exit</code>	Returns to global configuration mode.
Step 7	template port-channel <i>channel-name</i> Example: <code>apicl(config)# template port-channel corp</code>	Creates a new port-channel or configures an existing port-channel (global configuration). All VPCs 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 8	vlan-domain member <i>vlan-domain-name</i> Example: <code>vlan-domain member dom1</code>	Associates the port channel template with the previously configured VLAN domain.
Step 9	switchport access vlan <i>vlan-id</i> tenant <i>tenant-name</i> application <i>application-name</i> epg <i>epg-name</i> Example: <code>apicl(config-po-ch-if)# switchport access vlan 4 tenant ExampleCorp application Web epg webEpg</code>	Deploys the EPG with the VLAN on all ports with which the port-channel is associated.
Step 10	channel-mode active Example: <code>apicl(config-po-ch-if)# channel-mode active</code>	Note A port-channel must be in active channel-mode for a VPC.
Step 11	exit Example: <code>apicl(config-po-ch-if)# exit</code>	Returns to configure mode.
Step 12	leaf <i>node-id1 node-id2</i> Example: <code>apicl(config)# leaf 101-102</code>	Specifies the pair of leaf switches to be configured.

	Command or Action	Purpose
Step 13	interface <i>type leaf/interface-range</i> Example: <code>apicl(config-leaf)# interface ethernet 1/3-4</code>	Specifies the interface or range of interfaces that you are configuring to the port-channel.
Step 14	[no] channel-group <i>channel-name vpc</i> Example: <code>apicl(config-leaf-if)# channel-group corp vpc</code>	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 15	exit Example: <code>apicl(config-leaf-if)# exit</code>	
Step 16	exit Example: <code>apicl(config-leaf)# exit</code>	
Step 17	vpc context leaf <i>node-id1 node-id2</i> Example: <code>apicl(config)# vpc context leaf 101 102</code>	The vpc context mode allows configuration of VPC to be applied to both leaf node pairs.
Step 18	interface vpc <i>channel-name</i> Example: <code>apicl(config-vpc)# interface vpc blue fex 102 102</code>	
Step 19	(Optional) [no] shutdown Example: <code>apicl(config-vpc-if)# no shut</code>	Administrative state configuration in the vpc context allows changing the admin state of a VPC with one command for both leaf nodes.

Example

This example shows how to configure a basic VPC.

```

apicl# configure
apicl(config)# vlan-domain dom1 dynamic

apicl(config-vlan)# vlan 1000-1999
apicl(config-vlan)# exit
apicl(config)# vpc domain explicit 1 leaf 101 102
apicl(config-vpc)# peer-dead-interval 10

apicl(config-vpc)# exit
apicl(config)# template port-channel corp
apicl(config-po-ch-if)# vlan-domain member dom1

apicl(config-po-ch-if)# channel-mode active

apicl(config-po-ch-if)# exit
apicl(config)# leaf 101-102
apicl(config-leaf)# interface ethernet 1/3-4
apicl(config-leaf-if)# channel-group corp vpc
apicl(config-leaf-if)# exit
apicl(config)# vpc context leaf 101 102

```

This example shows how to configure VPCs with FEX ports.

```

apicl(config-leaf)# interface ethernet 101/1/1-2
apicl(config-leaf-if)# channel-group Reg vpc
apicl(config)# vpc context leaf 101 102
apicl(config-vpc)# interface vpc corp
apicl(config-vpc-if)# exit
apicl(config-vpc)# interface vpc red fex 101 101
apicl(config-vpc-if)# switchport
apicl(config-vpc-if)# exit
apicl(config-vpc)# interface vpc blue fex 102 102
apicl(config-vpc-if)# shut

```

Configuring Virtual Port Channels Using the REST API

Configuring a Single Virtual Port Channel Across Two Switches Using the REST API

The two steps for creating a virtual port channel across two switches are as follows:

- Create a `fabricExplicitGep`: this policy specifies the leaf switch that pairs to form the virtual port channel.
- Use the `infra` selector to specify the interface configuration.

The APIC performs several validations of the `fabricExplicitGep` and faults are raised when any of these validations fail. A leaf can be paired with only one other leaf. The APIC rejects any configuration that breaks this rule. When creating a `fabricExplicitGep`, an administrator must provide the IDs of both of the leaf switches to be paired. The APIC rejects any configuration which breaks this rule. Both switches must be up when `fabricExplicitGep` is created. If one switch is not up, the APIC accepts the configuration but raises a fault. Both switches must be leaf switches. If one or both switch IDs corresponds to a spine, the APIC accepts the configuration but raises a fault.

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 and protocol(s) are configured and available.

Procedure

To create the `fabricExplicitGep` policy and use the `intra` selector to specify the interface, send a post with XML such as the following example:

Example:

```
<fabricProtPol pairT="explicit">
<fabricExplicitGep name="tG" id="2">
  <fabricNodePEp id="18"/>
  <fabricNodePEp id="25"/>
    </fabricExplicitGep>
</fabricProtPol>
```

Configuring a Virtual Port Channel on Selected Port Blocks of Two Switches Using the REST API

This policy creates a single virtual port channel (VPC) on leaf switches 18 and 25, using interfaces 1/10 to 1/15 on leaf 18, and interfaces 1/20 to 1/25 on leaf 25.

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 and protocol(s) are configured 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

To create the VPC send a post with XML such as the following example:

Example:

```
<infraInfra dn="uni/infra">

  <infraNodeP name="test1">
    <infraLeafS name="leafs" type="range">
      <infraNodeBlk name="nblk"
        from_"18" to_"18"/>
    </infraLeafS>
    <infraRsAccPortP tDn="uni/infra/accportprof-test1"/>
  </infraNodeP>

  <infraNodeP name="test2">
    <infraLeafS name="leafs" type="range">
      <infraNodeBlk name="nblk"
        from_"25" to_"25"/>
    </infraLeafS>
    <infraRsAccPortP tDn="uni/infra/accportprof-test2"/>
  </infraNodeP>

  <infraAccPortP name="test1">
    <infraHPortS name="pselc" type="range">
      <infraPortBlk name="blk1"
        fromCard="1" toCard="1"
        fromPort="10" toPort="15"/>
      <infraRsAccBaseGrp
        tDn="uni/infra/funcprof/accbundle-bndlgrp" />
    </infraHPortS>
  </infraAccPortP>

  <infraAccPortP name="test2">
    <infraHPortS name="pselc" type="range">
      <infraPortBlk name="blk1"
        fromCard="1" toCard="1"
        fromPort="20" toPort="25"/>
      <infraRsAccBaseGrp
        tDn="uni/infra/funcprof/accbundle-bndlgrp" />
    </infraHPortS>
  </infraAccPortP>

  <infraFuncP>
    <infraAccBndlGrp name="bndlgrp" lagT="node">
      <infraRsHIfPol tnFabricHIfPolName="default"/>
      <infraRsCdpIfPol tnCdpIfPolName="default"/>
      <infraRsLacpPol tnLacpLagPolName="default"/>
    </infraAccBndlGrp>
  </infraFuncP>

</infraInfra>
```

Configuring a Single Virtual Port Channel Across Two Switches Using the REST API

The two steps for creating a virtual port channel across two switches are as follows:

- Create a `fabricExplicitGep`: this policy specifies the leaf switch that pairs to form the virtual port channel.

- Use the infra selector to specify the interface configuration.

The APIC performs several validations of the `fabricExplicitGep` and faults are raised when any of these validations fail. A leaf can be paired with only one other leaf. The APIC rejects any configuration that breaks this rule. When creating a `fabricExplicitGep`, an administrator must provide the IDs of both of the leaf switches to be paired. The APIC rejects any configuration which breaks this rule. Both switches must be up when `fabricExplicitGep` is created. If one switch is not up, the APIC accepts the configuration but raises a fault. Both switches must be leaf switches. If one or both switch IDs corresponds to a spine, the APIC accepts the configuration but raises a fault.

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 and protocol(s) are configured and available.

Procedure

To create the `fabricExplicitGep` policy and use the intra selector to specify the interface, send a post with XML such as the following example:

Example:

```
<fabricProtPol pairT="explicit">
<fabricExplicitGep name="tG" id="2">
  <fabricNodePEp id="18"/>
  <fabricNodePEp id="25"/>
    </fabricExplicitGep>
</fabricProtPol>
```

FEX Interfaces

Configuring Port, PC, and VPC Connections to FEX Devices

FEX connections and the profiles used to configure them can be created using the GUI, NX-OS Style CLI, or the REST API.

For information on how to configure them using the NX-OS style CLI, see the topics about configuring ports, PCs and VPCs using the NX-OS style CLI.

ACI FEX Guidelines

Observe the following guidelines when deploying a FEX:

- Assuming that no leaf switch front panel ports are configured to deploy and EPG and VLANs, a maximum of 10,000 port EPGs are supported for being deployed using a FEX.

- For each FEX port or vPC that includes FEX ports as members, a maximum of 20 EPGs per VLAN are supported.

FEX Virtual Port Channels

As of the v1.1 release, the ACI fabric supports Cisco Fabric Extender (FEX) server-side virtual port channels (VPC), also known as an FEX straight-through VPC.

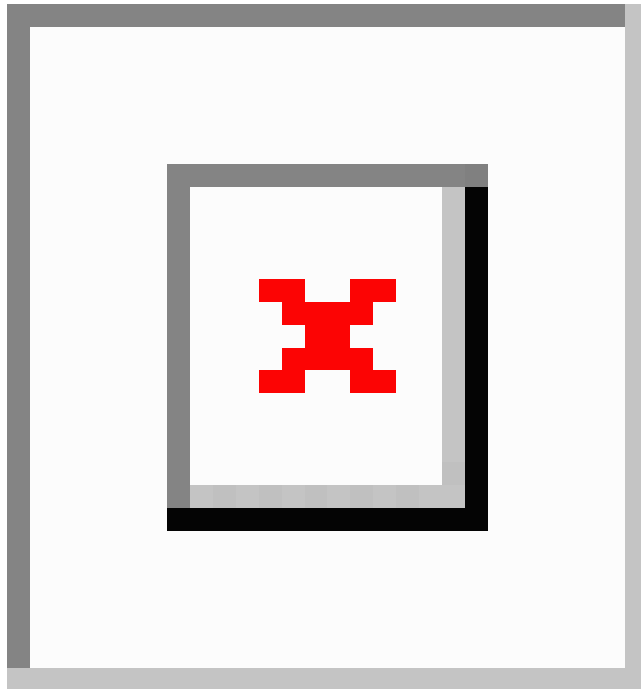


Note When creating a VPC domain between two leaf switches, both switches must be in the same switch generation, one of the following:

- 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.

Figure 5: Supported FEX VPC Topologies



Supported FEX VPC port channel topologies include the following:

- Both VTEP and non-VTEP hypervisors behind a FEX.
- Virtual switches (such as AVS or VDS) connected to two FEXs that are connected to the ACI fabric (VPCs directly connected on physical FEX ports is not supported - a VPC is supported only on port channels).



Note When using GARP as the protocol to notify of IP to Mac binding changes to different interfaces on the same FEX requires the bridge domain mode is set to flood and GARP trap to CPU is not enabled.

Configuring a Basic FEX Connection Using the GUI

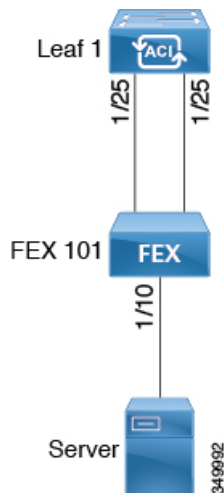


Note Cisco Discovery Protocol (CDP) is not supported with FEX switches. Instead, use LLDP for neighbor discovery.

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 6: 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 switches, interfaces, and protocol(s) are configured and available.
- The FEX is powered on and connected to the target leaf interfaces



Note A maximum of eight members is supported in fabric port-channels connected to FEXs.

Procedure

Step 1 On the APIC, create a switch profile using the **Fabric > Access Policies > Quick Start Configure 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 profile to the APIC. The APIC auto-generates the necessary FEX profile (*<switch policy name>_FexP<FEX ID>*) and selector (*<switch policy name>_ifselector*).

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-generated 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 profile 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.

This completes the basic FEX configuration steps.

What to do next



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 FEX Port Channel Connections Using the GUI

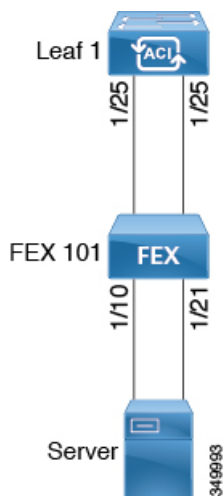


Note Cisco Discovery Protocol (CDP) is not supported with FEX switches. Instead, use LLDP for neighbor discovery.

The main steps are as follows:

1. Configure an FEX profile to use FEX ports to form a port channel.
2. Configure the port channel to enable attaching a server.

Figure 7: 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.

- On the APIC menu bar, navigate to **Fabric > Access Policies > Switch Policies > Profiles**.
- In the **Navigation Pane**, select the FEX profile.

APIC auto-generated FEX profile names are formed as follows: *<switch policy name>_FexP<FEX ID>*.

- 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.

- Provide a name for the selector.
- Specify the FEX interface IDs to use.
- 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.

- In the *Port Channel Policy* option, select static or dynamic LACP according to the requirements of your configuration.
- Click **Submit** to submit the updated FEX profile 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**

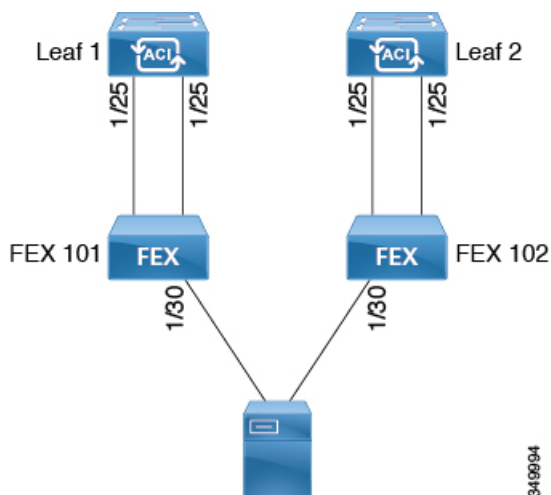
While this configuration enables hardware connectivity, no data traffic can flow without a valid application profile, EPG, and contract that is associated with this hardware configuration.

Configuring FEX VPC Connections Using the GUI

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 8: 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



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, 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.
CDP policies are not supported in interface policy groups used with FEX interfaces.
- 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 profile 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, 1 gbit 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 profile 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.

Configuring an FEX VPC Policy Using the REST API

This task creates a FEX virtual port channel (VPC) policy.

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



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

To create the policy linking the FEX through a VPC to two switches, send a post with XML such as the following example:

Example:

```
<polUni>
<infraInfra dn="uni/infra">

<infraNodeP name="fexNodeP105">
  <infraLeafS name="leafs" type="range">
    <infraNodeBlk name="test" from_"105" to_"105"/>
  </infraLeafS>
  <infraRsAccPortP tDn="uni/infra/accportprof-fex116nif105" />
</infraNodeP>

<infraNodeP name="fexNodeP101">
  <infraLeafS name="leafs" type="range">
    <infraNodeBlk name="test" from_"101" to_"101"/>
  </infraLeafS>
  <infraRsAccPortP tDn="uni/infra/accportprof-fex113nif101" />
</infraNodeP>

<infraAccPortP name="fex116nif105">
  <infraHPortS name="pselc" type="range">
    <infraPortBlk name="blk1"
      fromCard="1" toCard="1" fromPort="45" toPort="48" >
    </infraPortBlk>
    <infraRsAccBaseGrp tDn="uni/infra/fexprof-fexHIF116/fexbundle-fex116" fexId="116" />
  </infraHPortS>
</infraAccPortP>

<infraAccPortP name="fex113nif101">
  <infraHPortS name="pselc" type="range">
    <infraPortBlk name="blk1"
      fromCard="1" toCard="1" fromPort="45" toPort="48" >
    </infraPortBlk>
    <infraRsAccBaseGrp tDn="uni/infra/fexprof-fexHIF113/fexbundle-fex113" fexId="113" />
  </infraHPortS>
</infraAccPortP>

<infraFexP name="fexHIF113">
  <infraFexBndlGrp name="fex113"/>
  <infraHPortS name="pselc-fexPC" type="range">
    <infraPortBlk name="blk"

```

```

        fromCard="1" toCard="1" fromPort="15" toPort="16" >
    </infraPortBlk>
    <infraRsAccBaseGrp tDn="uni/infra/funcprof/accbundle-fexPCbundle" />
</infraHPortS>
    <infraHPortS name="pselc-fexVPC" type="range">
    <infraPortBlk name="blk"
        fromCard="1" toCard="1" fromPort="1" toPort="8" >
    </infraPortBlk>
    <infraRsAccBaseGrp tDn="uni/infra/funcprof/accbundle-fexvpcbundle" />
</infraHPortS>
<infraHPortS name="pselc-fexaccess" type="range">
    <infraPortBlk name="blk"
        fromCard="1" toCard="1" fromPort="47" toPort="47">
    </infraPortBlk>
    <infraRsAccBaseGrp tDn="uni/infra/funcprof/accportgrp-fexaccport" />
</infraHPortS>

</infraFexP>

<infraFexP name="fexHIF116">
    <infraFexBndlGrp name="fex116"/>
    <infraHPortS name="pselc-fexPC" type="range">
    <infraPortBlk name="blk"
        fromCard="1" toCard="1" fromPort="17" toPort="18" >
    </infraPortBlk>
    <infraRsAccBaseGrp tDn="uni/infra/funcprof/accbundle-fexPCbundle" />
</infraHPortS>
    <infraHPortS name="pselc-fexVPC" type="range">
    <infraPortBlk name="blk"
        fromCard="1" toCard="1" fromPort="1" toPort="8" >
    </infraPortBlk>
    <infraRsAccBaseGrp tDn="uni/infra/funcprof/accbundle-fexvpcbundle" />
</infraHPortS>
    <infraHPortS name="pselc-fexaccess" type="range">
    <infraPortBlk name="blk"
        fromCard="1" toCard="1" fromPort="47" toPort="47">
    </infraPortBlk>
    <infraRsAccBaseGrp tDn="uni/infra/funcprof/accportgrp-fexaccport" />
</infraHPortS>

</infraFexP>

<infraFuncP>
<infraAccBndlGrp name="fexPCbundle" lagT="link">
    <infraRsLacpPol tnLacpLagPolName='staticLag' />
    <infraRsHifPol tnFabricHifPolName="1GHifPol" />
    <infraRsAttEntP tDn="uni/infra/attentp-fexvpcAttEP"/>
</infraAccBndlGrp>

<infraAccBndlGrp name="fexvpcbundle" lagT="node">
    <infraRsLacpPol tnLacpLagPolName='staticLag' />
    <infraRsHifPol tnFabricHifPolName="1GHifPol" />
    <infraRsAttEntP tDn="uni/infra/attentp-fexvpcAttEP"/>
</infraAccBndlGrp>
</infraFuncP>

<fabricHifPol name="1GHifPol" speed="1G" />
<infraAttEntityP name="fexvpcAttEP">
    <infraProvAcc name="provfunc"/>
    <infraRsDomP tDn="uni/phys-fexvpcDOM"/>
</infraAttEntityP>

```

```
<lacpLagPol dn="uni/infra/lacplagp-staticLag"  
  ctrl="susp-individual,graceful-conv"  
  minLinks="2"  
  maxLinks="16">  
</lacpLagPol>
```
