



# Configuring Virtual Port Channels

---

This chapter contains the following sections:

- [Information About vPCs, page 1](#)
- [Guidelines and Limitations for vPCs, page 15](#)
- [Configuring vPCs, page 15](#)
- [Configuring the vPC Peer Switch, page 28](#)
- [Verifying the vPC Configuration, page 31](#)
- [vPC Example Configurations, page 36](#)
- [vPC Default Settings, page 40](#)

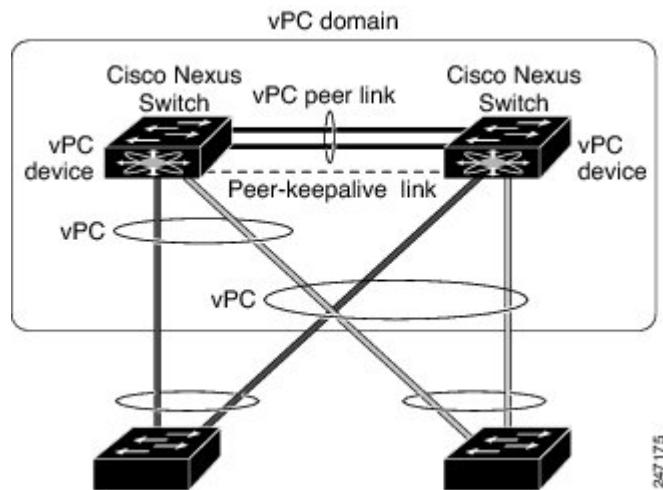
## Information About vPCs

### vPC Overview

A virtual port channel (vPC) allows links that are physically connected to two different Cisco Nexus devices or Cisco Nexus Fabric Extenders to appear as a single port channel by a third device (see the following figure). The third device can be a switch, server, or any other networking device. You can configure vPCs in topologies that include Cisco Nexus devices connected to Cisco Nexus Fabric Extenders. A vPC can provide multipathing,

which allows you to create redundancy by enabling multiple parallel paths between nodes and load balancing traffic where alternative paths exist.

**Figure 1: vPC Architecture**



You configure the EtherChannels by using one of the following:

- No protocol
- Link Aggregation Control Protocol (LACP)

When you configure the EtherChannels in a vPC—including the vPC peer link channel—each switch can have up to 16 active links in a single EtherChannel. When you configure a vPC on a Fabric Extender, only one port is allowed in an EtherChannel.



**Note**

You must enable the vPC feature before you can configure or run the vPC functionality.

To enable the vPC functionality, you must create a peer-keepalive link and a peer-link under the vPC domain for the two vPC peer switches to provide the vPC functionality.

To create a vPC peer link you configure an EtherChannel on one Cisco Nexus device by using two or more Ethernet ports. On the other switch, you configure another EtherChannel again using two or more Ethernet ports. Connecting these two EtherChannels together creates a vPC peer link.



**Note**

We recommend that you configure the vPC peer-link EtherChannels as trunks.

The vPC domain includes both vPC peer devices, the vPC peer-keepalive link, the vPC peer link, and all of the EtherChannels in the vPC domain connected to the downstream device. You can have only one vPC domain ID on each vPC peer device.



**Note**

Always attach all vPC devices using EtherChannels to both vPC peer devices.

A vPC provides the following benefits:

- Allows a single device to use an EtherChannel across two upstream devices
- Eliminates Spanning Tree Protocol (STP) blocked ports
- Provides a loop-free topology
- Uses all available uplink bandwidth
- Provides fast convergence if either the link or a switch fails
- Provides link-level resiliency
- Assures high availability

## Terminology

### vPC Terminology

The terminology used in vPCs is as follows:

- vPC—combined EtherChannel between the vPC peer devices and the downstream device.
- vPC peer device—One of a pair of devices that are connected with the special EtherChannel known as the vPC peer link.
- vPC peer link—link used to synchronize states between the vPC peer devices.
- vPC member port—Interfaces that belong to the vPCs.
- Host vPC port—Fabric Extender host interfaces that belong to a vPC.
- vPC domain—domain that includes both vPC peer devices, the vPC peer-keepalive link, and all of the port channels in the vPC connected to the downstream devices. It is also associated to the configuration mode that you must use to assign vPC global parameters. The vPC domain ID must be the same on both switches.
- vPC peer-keepalive link—The peer-keepalive link monitors the vitality of a vPC peer Cisco Nexus device. The peer-keepalive link sends configurable, periodic keepalive messages between vPC peer devices.

No data or synchronization traffic moves over the vPC peer-keepalive link; the only traffic on this link is a message that indicates that the originating switch is operating and running vPCs.

### Fabric Extender Terminology

The terminology used for the Cisco Nexus Fabric Extender is as follows:

- Fabric interface—A 10-Gigabit Ethernet uplink port designated for connection from the Fabric Extender to its parent switch. A fabric interface cannot be used for any other purpose. It must be directly connected to the parent switch.
- EtherChannel fabric interface—An EtherChannel uplink connection from the Fabric Extender to its parent switch. This connection consists of fabric interfaces bundled into a single logical channel.

- Host interface—An Ethernet interface for server or host connectivity. These ports are 1-Gigabit Ethernet interfaces or 10-Gigabit Ethernet interfaces, depending on the fabric extender model.
- EtherChannel host interface—An EtherChannel downlink connection from the Fabric Extender host interface to a server port.



**Note** An EtherChannel host interface consists of only one host interface and can be configured either as a Link Aggregation Control Protocol (LACP) or non-LACP EtherChannel.

## Supported vPC Topologies

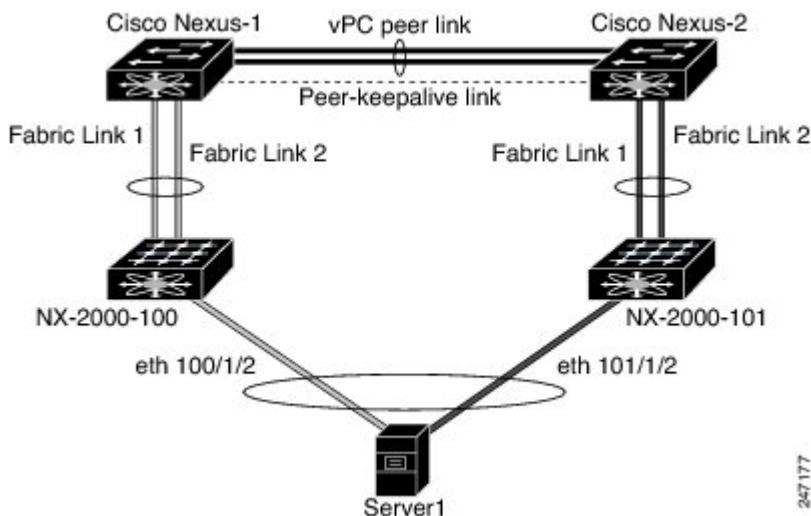
### Cisco Nexus Device vPC Topology

#### Single Homed Fabric Extender vPC Topology

You can connect a server with dual or quad or more network adapters that are configured in a vPC to a pair of Cisco Nexus Fabric Extenders which are connected to the Cisco Nexus devices as depicted. Depending on the FEX model, you may be able to connect one or more network adapter interfaces to each fabric extender. As an example, the following figure refers to a topology built with the Cisco Nexus 2148T fabric extender, where a server has one link only to each fabric extender. A topology with Cisco Nexus 2248TP or with Cisco Nexus 2232PP fabric extender could consist of more links from the server to a single fabric extender.

- . The topology that is shown in the following figure provides the vPC functionality to dual homed servers with 1-Gigabit Ethernet uplink interfaces.

**Figure 2: Single Homed Fabric Extender vPC Topology**



The Cisco Nexus device can support up to 12 configured single homed Fabric Extenders (576 ports) with this topology however only 480 576 dual homed host servers can be configured in a vPCs with this configuration.

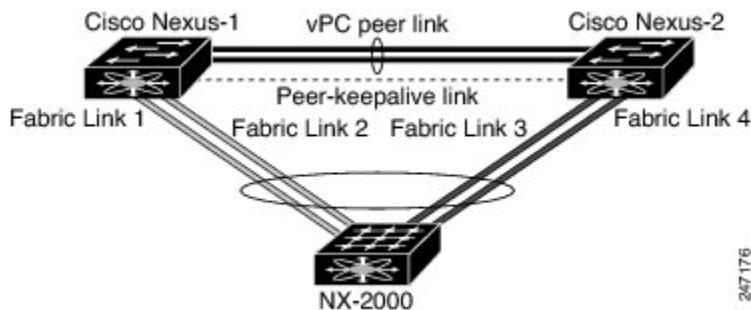
**Note**

The Cisco Nexus 2148T fabric extender does not support EtherChannels on its host interfaces. Therefore a maximum of two links can be configured in an EtherChannel from the server where each link is connected to a separate Fabric Extender.

## Dual Homed Fabric Extender vPC Topology

You can connect the Cisco Nexus Fabric Extender to two upstream Cisco Nexus devices and downstream to a number of single homed servers. The topology shown in the following figure provides the vPC functionality to singly connected servers with 1-Gigabit Ethernet uplink interfaces.

**Figure 3: Dual Homed Fabric Extender vPC Topology**



The Cisco Nexus device can support up to 12 configured dual homed Fabric Extenders with this topology. A maximum of 576 single homed servers can be connected to this configuration.

## vPC Domain

To create a vPC domain, you must first create a vPC domain ID on each vPC peer switch using a number from 1 to 1000. This ID must be the same on a set of vPC peer devices.

You can configure the EtherChannels and vPC peer links by using LACP or no protocol. When possible, we recommend that you use LACP on the peer-link, because LACP provides configuration checks against a configuration mismatch on the EtherChannel.

The vPC peer switches use the vPC domain ID that you configure to automatically assign a unique vPC system MAC address. Each vPC domain has a unique MAC address that is used as a unique identifier for the specific vPC-related operations, although the switches use the vPC system MAC addresses only for link-scope operations, such as LACP. We recommend that you create each vPC domain within the contiguous network with a unique domain ID. You can also configure a specific MAC address for the vPC domain, rather than having the Cisco NX-OS software assign the address.

The vPC peer switches use the vPC domain ID that you configure to automatically assign a unique vPC system MAC address. The switches use the vPC system MAC addresses only for link-scope operations, such as LACP or BPDUs. You can also configure a specific MAC address for the vPC domain.

We recommend that you configure the same VPC domain ID on both peers and, the domain ID should be unique in the network. For example, if there are two different VPCs (one in access and one in aggregation) then each vPC should have a unique domain ID.

After you create a vPC domain, the Cisco NX-OS software automatically creates a system priority for the vPC domain. You can also manually configure a specific system priority for the vPC domain.

**Note**

If you manually configure the system priority, you must ensure that you assign the same priority value on both vPC peer switches. If the vPC peer switches have different system priority values, the vPC will not come up.

## Peer-Keepalive Link and Messages

The Cisco NX-OS software uses a peer-keepalive link between the vPC peers to transmit periodic, configurable keepalive messages. You must have Layer 3 connectivity between the peer switches to transmit these messages; the system cannot bring up the vPC peer link unless a peer-keepalive link is already up and running.

You can configure a hold-timeout and a timeout value simultaneously.

**Hold-timeout value**—The hold-timeout value range is between 3 to 10 seconds, with a default value of 3 seconds. This timer starts when the vPC peer link goes down. The purpose of the hold-timeout period is to prevent false-positive cases.

If you configure a hold-timeout value that is lower than the timeout value, then the vPC system ignores vPC peer-keepalive messages for the hold-timeout period and considers messages for the remainder of the timeout period. If no keepalive message is received for this period, the vPC secondary device takes over the role of the primary device. For example, if the hold-timeout value is 3 seconds and the timeout value is 5 seconds, for the first 3 seconds vPC keepalive messages are ignored (such as, when accommodating a supervisor failure for a few seconds after peer link failure) and keepalive messages are considered for the remaining timeout period of 2 seconds. After this period, the vPC secondary device takes over as the primary device, in case there is no keep alive message.

**Timeout value**—The timeout value range is between 3 to 20 seconds, with a default value of 5 seconds. This timer starts at the end of the hold-timeout interval. If you configure a timeout value that is lower than or equal to the hold-timeout value, then the timeout duration is initiated after the hold-timeout period. For example, if the timeout value is 3 seconds and the hold-timeout value is 5 seconds, the timeout period starts after 5 seconds.

**Note**

We recommend that you configure the vPC peer-keepalive link on the Cisco Nexus device to run in the management VRF using the mgmt 0 interfaces. If you configure the default VRF, ensure that the vPC peer link is not used to carry the vPC peer-keepalive messages.

## Compatibility Parameters for vPC Peer Links

Many configuration and operational parameters must be identical on all interfaces in the vPC. After you enable the vPC feature and configure the peer link on both vPC peer switches, Cisco Fabric Services (CFS) messages provide a copy of the configuration on the local vPC peer switch configuration to the remote vPC peer switch. The system then determines whether any of the crucial configuration parameters differ on the two switches.

Enter the **show vpc consistency-parameters** command to display the configured values on all interfaces in the vPC. The displayed configurations are only those configurations that would limit the vPC peer link and vPC from coming up.

The compatibility check process for vPCs differs from the compatibility check for regular EtherChannels.

## Configuration Parameters That Must Be Identical

The configuration parameters in this section must be configured identically on both switches at either end of the vPC peer link.

**Note**

You must ensure that all interfaces in the vPC have the identical operational and configuration parameters listed in this section.

Enter the **show vpc consistency-parameters** command to display the configured values on all interfaces in the vPC. The displayed configurations are only those configurations that would limit the vPC peer link and vPC from coming up.

---

The switch automatically checks for compatibility of these parameters on the vPC interfaces. The per-interface parameters must be consistent per interface, and the global parameters must be consistent globally.

- Port-channel mode: on, off, or active
- Link speed per channel
- Duplex mode per channel
- Trunk mode per channel:
  - Native VLAN
  - VLANs allowed on trunk
  - Tagging of native VLAN traffic
- Spanning Tree Protocol (STP) mode
- STP region configuration for Multiple Spanning Tree (MST)
- Enable or disable state per VLAN
- STP global settings:
  - Bridge Assurance setting
  - Port type setting—We recommend that you set all vPC interfaces as normal ports
  - Loop Guard settings
- STP interface settings:
  - Port type setting
  - Loop Guard
  - Root Guard
- For the Fabric Extender vPC topology, all the interface level parameters mentioned above should be identically configured for host interface from both the switches.

- Fabric Extender FEX number configured on an EtherChannel fabric interface; for the Fabric Extender vPC topology.

If any of these parameters are not enabled or defined on either switch, the vPC consistency check ignores those parameters.


**Note**

To ensure that none of the vPC interfaces are in the suspend mode, enter the **show vpc brief** and **show vpc consistency-parameters** commands and check the syslog messages.

## Configuration Parameters That Should Be Identical

When any of the following parameters are not configured identically on both vPC peer switches, a misconfiguration might cause undesirable behavior in the traffic flow:

- MAC aging timers
- Static MAC entries
- VLAN interface—Each switch on the end of the vPC peer link must have a VLAN interface configured for the same VLAN on both ends and they must be in the same administrative and operational mode. Those VLANs configured on only one switch of the peer link do not pass traffic using the vPC or peer link. You must create all VLANs on both the primary and secondary vPC switches, or the VLAN will be suspended.
- Private VLAN configuration
- All ACL configurations and parameters
- Quality of service (QoS) configuration and parameters—Local parameters; global parameters must be identical
- STP interface settings:
  - BPDU Filter
  - BPDU Guard
  - Cost
  - Link type
  - Priority
  - VLANs (Rapid PVST+)

To ensure that all the configuration parameters are compatible, we recommend that you display the configurations for each vPC peer switch once you configure the vPC.

## Graceful Type-1 Check

When a consistency check fails, vPCs are brought down only on the secondary vPC switch. The VLANs remain up on the primary switch and Type-1 configurations can be performed without traffic disruption. This feature is used both in the case of global as well as interface-specific Type-1 inconsistencies.

This feature is not enabled for dual-active FEX ports. When a Type-1 mismatch occurs, VLANs are suspended on these ports on both switches.

## Per-VLAN Consistency Check

Beginning with Cisco NX-OS Release 5.0(2)N2(1), some Type-1 consistency checks are performed on a per-VLAN basis when spanning tree is enabled or disabled on a VLAN. VLANs that do not pass the consistency check are brought down on both the primary and secondary switches while other VLANs are not affected.

## vPC Auto-Recovery

Beginning with Cisco NX-OS Release 5.0(2)N2(1), the vPC auto-recovery feature re-enables vPC links in the following scenarios:

When both vPC peer switches reload and only one switch reboots, auto-recovery allows that switch to assume the role of the primary switch and the vPC links will be allowed to come up after a predetermined period of time. The reload delay period in this scenario can range from 240 to 3600 seconds.

When vPCs are disabled on a secondary vPC switch due to a peer-link failure and then the primary vPC switch fails or is unable to forward traffic, the secondary switch reenables the vPCs. In this scenario, the vPC waits for three consecutive keepalive failures to recover the vPC links.

## vPC Peer Links

A vPC peer link is the link that is used to synchronize the states between the vPC peer devices.

**Note**

You must configure the peer-keepalive link before you configure the vPC peer link or the peer link will not come up.

## vPC Peer Link Overview

You can have only two switches as vPC peers; each switch can serve as a vPC peer to only one other vPC peer. The vPC peer switches can also have non-vPC links to other switches.

To make a valid configuration, you configure an EtherChannel on each switch and then configure the vPC domain. You assign the EtherChannel on each switch as a peer link. For redundancy, we recommend that you should configure at least two dedicated ports into the EtherChannel; if one of the interfaces in the vPC peer link fails, the switch automatically falls back to use another interface in the peer link.

**Note**


---

We recommend that you configure the EtherChannels in trunk mode.

---

Many operational parameters and configuration parameters must be the same in each switch connected by a vPC peer link. Because each switch is completely independent on the management plane, you must ensure that the switches are compatible on the critical parameters. vPC peer switches have separate control planes. After configuring the vPC peer link, you should display the configuration on each vPC peer switch to ensure that the configurations are compatible.

**Note**


---

You must ensure that the two switches connected by the vPC peer link have certain identical operational and configuration parameters.

---

When you configure the vPC peer link, the vPC peer switches negotiate that one of the connected switches is the primary switch and the other connected switch is the secondary switch. By default, the Cisco NX-OS software uses the lowest MAC address to elect the primary switch. The software takes different actions on each switch—that is, the primary and secondary—only in certain failover conditions. If the primary switch fails, the secondary switch becomes the operational primary switch when the system recovers, and the previously primary switch is now the secondary switch.

You can also configure which of the vPC switches is the primary switch. If you want to configure the role priority again to make one vPC switch the primary switch, configure the role priority on both the primary and secondary vPC switches with the appropriate values, shut down the EtherChannel that is the vPC peer link on both switches by entering the **shutdown** command, and reenable the EtherChannel on both switches by entering the **no shutdown** command.

MAC addresses that are learned over vPC links are also synchronized between the peers.

Configuration information flows across the vPC peer links using the Cisco Fabric Services over Ethernet (CFSoE) protocol. All MAC addresses for those VLANs configured on both switches are synchronized between vPC peer switches. The software uses CFSoE for this synchronization.

If the vPC peer link fails, the software checks the status of the remote vPC peer switch using the peer-keepalive link, which is a link between vPC peer switches, to ensure that both switches are up. If the vPC peer switch is up, the secondary vPC switch disables all vPC ports on its switch. The data then forwards down the remaining active links of the EtherChannel.

The software learns of a vPC peer switch failure when the keepalive messages are not returned over the peer-keepalive link.

Use a separate link (vPC peer-keepalive link) to send configurable keepalive messages between the vPC peer switches. The keepalive messages on the vPC peer-keepalive link determines whether a failure is on the vPC peer link only or on the vPC peer switch. The keepalive messages are used only when all the links in the peer link fail.

## vPC Number

Once you have created the vPC domain ID and the vPC peer link, you can create EtherChannels to attach the downstream switch to each vPC peer switch. That is, you create one single EtherChannel on the downstream switch with half of the ports to the primary vPC peer switch and the other half of the ports to the secondary peer switch.

On each vPC peer switch, you assign the same vPC number to the EtherChannel that connects to the downstream switch. You will experience minimal traffic disruption when you are creating vPCs. To simplify the configuration, you can assign the vPC ID number for each EtherChannel to be the same as the EtherChannel itself (that is, vPC ID 10 for EtherChannel 10).

**Note**

The vPC number that you assign to the EtherChannel that connects to the downstream switch from the vPC peer switch must be identical on both vPC peer switches.

## vPC Interactions with Other Features

### Configuring vPC Peer Links and Links to the Core

Configure the command line interface by using a track object and a track list that is associated with the Layer 3 link to the core and on all vPC peer links on both vPC peer devices. You use this configuration to avoid dropping traffic if that particular module goes down because when all the tracked objects on the track list go down, the system does the following:

- Stops the vPC primary peer device sending peer-keepalive messages which forces the vPC secondary peer device to take over.
- Brings down all the downstream vPCs on that vPC peer device, which forces all the traffic to be rerouted in the access switch toward the other vPC peer device.

Once you configure this feature and if the module fails, the system automatically suspends all the vPC links on the primary vPC peer device and stops the peer-keepalive messages. This action forces the vPC secondary device to take over the primary role and all the vPC traffic to go to this new vPC primary device until the system stabilizes.

Create a track list that contains all the links to the core and all the vPC peer links as its object. Enable tracking for the specified vPC domain for this track list. Apply this same configuration to the other vPC peer device.

#### Before You Begin

To configure a track list to switch over vPC to the remote peer when all related interfaces fail:

#### Procedure

	<b>Command or Action</b>	<b>Purpose</b>
<b>Step 1</b>	switch# <b>configure terminal</b>	Enters global configuration mode.
<b>Step 2</b>	switch(config)# <b>interface type slot/port</b>	Enters interface configuration mode.
<b>Step 3</b>	switch(config-if)# <b>track track-id interface type slot/port line-protocol</b>	Configures the track objects on an interface (Layer 3 to core).
<b>Step 4</b>	switch(config-track)# <b>track track-id interface type slot/port line-protocol</b>	Tracks the objects on an interface (Layer 3 to core).

	<b>Command or Action</b>	<b>Purpose</b>
<b>Step 5</b>	switch(config)# <b>track track-id interface port-channel port line-protocol</b>	Configures the track objects on a port channel (vPC peer link).
<b>Step 6</b>	switch(config)# <b>track track-id list boolean [OR   AND]</b>	Creates a track list that contains all the interfaces in the track list using the Boolean OR to trigger when all the objects fail, or trigger a switchover when any core interface or peer-link goes down using Boolean AND.
<b>Step 7</b>	switch(config-track)# <b>object number</b>	Specifies the object number.
<b>Step 8</b>	switch(config-track)# <b>end</b>	Exits track configuration mode.
<b>Step 9</b>	switch(config)# <b>vpc domain domain-id</b>	Enters vPC domain configuration.
<b>Step 10</b>	switch(config-vpc-domain)# <b>track number</b>	Adds the track object to the vPC domain.
<b>Step 11</b>	switch(config)# <b>show vpc brief</b>	(Optional) Displays the track object.
<b>Step 12</b>	switch(config)# <b>copy running-config startup-config</b>	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

This example shows how to configure a track list to trigger when all the objects fail using Boolean OR:

```
switch# configure terminal
switch(config)# interface ethernet 8/35
switch(config-if)# track 35 interface ethernet 8/35 line-protocol
switch(config-track)# track 23 interface ethernet 8/33 line-protocol
switch(config)# track 55 interface port-channel 100 line-protocol
switch(config)# track 44 list boolean OR
switch(config-track)# object 23
switch(config-track)# object 35
switch(config-track)# object 55
switch(config-track)# end
switch(config)# vpc domain 1
switch(config-vpc-domain)# track 44
switch(config)# copy running-config startup-config
```

## vPC and LACP

The Link Aggregation Control Protocol (LACP) uses the system MAC address of the vPC domain to form the LACP Aggregation Group (LAG) ID for the vPC.

You can use LACP on all the vPC EtherChannels, including those channels from the downstream switch. We recommend that you configure LACP with active mode on the interfaces on each EtherChannel on the vPC peer switches. This configuration allows you to more easily detect compatibility between switches, unidirectional links, and multihop connections, and provides dynamic reaction to run-time changes and link failures.

The vPC peer link supports 16 EtherChannel interfaces.

**Note**

When you manually configure the system priority, you must ensure that you assign the same priority value on both vPC peer switches. If the vPC peer switches have different system priority values, vPC does not come up.

**Note**

During a vPC role change (where secondary becomes primary), if the ports are in LACP-individual state on FEX, the ports will flap and this will cause brief connectivity loss before the primary is up. To avoid this issue, ensure that port-channel member ports are not configured as LACP-individual state or LACP is not used (On mode).

## vPC Peer Links and STP

When you first bring up the vPC functionality, STP reconverges. STP treats the vPC peer link as a special link and always includes the vPC peer link in the STP active topology.

We recommend that you set all the vPC peer link interfaces to the STP network port type so that Bridge Assurance is automatically enabled on all vPC peer links. We also recommend that you do not enable any of the STP enhancement features on VPC peer links.

You must configure a list of parameters to be identical on the vPC peer switches on both sides of the vPC peer link.

STP is distributed; that is, the protocol continues running on both vPC peer switches. However, the configuration on the vPC peer switch elected as the primary switch controls the STP process for the vPC interfaces on the secondary vPC peer switch.

The primary vPC switch synchronizes the STP state on the vPC secondary peer switch using Cisco Fabric Services over Ethernet (CFSoE).

The vPC manager performs a proposal/handshake agreement between the vPC peer switches that sets the primary and secondary switches and coordinates the two switches for STP. The primary vPC peer switch then controls the STP protocol for vPC interfaces on both the primary and secondary switches.

The Bridge Protocol Data Units (BPDUs) use the MAC address set for the vPC for the STP bridge ID in the designated bridge ID field. The vPC primary switch sends these BPDUs on the vPC interfaces.

**Note**

Display the configuration on both sides of the vPC peer link to ensure that the settings are identical. Use the **show spanning-tree** command to display information about the vPC.

## vPC and ARP

Table synchronization across vPC peers is managed in Cisco NX-OS using the reliable transport mechanism of the Cisco Fabric Services over Ethernet (CFSoE) protocol. To support faster convergence of address tables between the vPC peers, the **ip arp synchronize** command must be enabled. This convergence is designed to overcome the delay involved in ARP table restoration when the peer-link port channel flaps or when a vPC peer comes back online.

To improve performance, we recommend that you turn on the ARP sync feature. By default, it is not enabled.

To check whether or not ARP sync is enabled, enter the following command:

```
switch# show running
```

To enable ARP sync, enter the following command:

```
switch(config-vpc-domain) # ip arp synchronize
```

## CFSoE

The Cisco Fabric Services over Ethernet (CFSoE) is a reliable state transport mechanism that you can use to synchronize the actions of the vPC peer devices. CFSoE carries messages and packets for many features linked with vPC, such as STP and IGMP. Information is carried in CFS/CFSoE protocol data units (PDUs).

When you enable the vPC feature, the device automatically enables CFSoE, and you do not have to configure anything. CFSoE distributions for vPCs do not need the capabilities to distribute over IP or the CFS regions. You do not need to configure anything for the CFSoE feature to work correctly on vPCs.

You can use the **show mac address-table** command to display the MAC addresses that CFSoE synchronizes for the vPC peer link.



### Note

Do not enter the **no cfs eth distribute** or the **no cfs distribute** command. CFSoE must be enabled for vPC functionality. If you do enter either of these commands when vPC is enabled, the system displays an error message.

---

When you enter the **show cfs application** command, the output displays "Physical-eth," which shows the applications that are using CFSoE.

## vPC Peer Switch

The vPC peer switch feature addresses performance concerns around STP convergence. This feature allows a pair of Cisco Nexus devices to appear as a single STP root in the Layer 2 topology. This feature eliminates the need to pin the STP root to the vPC primary switch and improves vPC convergence if the vPC primary switch fails.

To avoid loops, the vPC peer link is excluded from the STP computation. In vPC peer switch mode, STP BPDUs are sent from both vPC peer devices to avoid issues related to STP BPDU timeout on the downstream switches, which can cause traffic disruption.

This feature can be used with the pure peer switch topology in which the devices all belong to the vPC.



### Note

---

Peer-switch feature is supported on networks that use vPC and STP-based redundancy is not supported. If the vPC peer-link fail in a hybrid peer-switch configuration, you can lose traffic. In this scenario, the vPC peers use the same STP root ID as well same bridge ID. The access switch traffic is split in two with half going to the first vPC peer and the other half to the second vPC peer. With the peer link failed, there is no impact on north/south traffic but east-west traffic will be lost (black-holed).

---

For information on STP enhancement features and Rapid PVST+, see the *Layer 2 Switching Configuration Guide* for your device.

# Guidelines and Limitations for vPCs

vPC has the following configuration guidelines and limitations:

- You must enable the vPC feature before you can configure vPC peer-link and vPC interfaces.
- You must configure the peer-keepalive link before the system can form the vPC peer link.
- The vPC peer-link needs to be formed using a minimum of two 10-Gigabit Ethernet interfaces.
- You can connect a pair of Cisco Nexus 5000 Series switches or a pair of Cisco Nexus 5500 Series switches in a vPC directly to another switch or to a server. vPC peer switches must be of the same type, for example, you can connect a pair of Nexus 5000 series switches or a pair of Nexus 5500 Series switches but you cannot connect a Nexus 5000 Series switch to a Nexus 5500 Series switch in a vPC topology.
- Only port channels can be in vPCs. A vPC can be configured on a normal port channel (switch-to-switch vPC topology), on a port channel fabric interface (fabric extender vPC topology), and on a port channel host interface (host interface vPC topology).
- A Fabric Extender can be a member of a Host Interface vPC topology or a Fabric Extender vPC topology but not both simultaneously.
- You must configure both vPC peer switches; the configuration is not automatically synchronized between the vPC peer devices.
- Check that the necessary configuration parameters are compatible on both sides of the vPC peer link.
- You may experience minimal traffic disruption while configuring vPCs.
- You should configure all the port channels in the vPC using LACP with the interfaces in active mode.
- When the **peer-switch** command is configured and vPC keepalive messages exchanged through an SVI instead of a management interface, additional Spanning Tree Protocol (STP) configuration is required. STP needs to be disabled on the dedicated link that carries the keepalive traffic between the vPC peers. You can disable STP on the dedicated link by configuring STP BPDUfilter on the both ends of the dedicated link. We recommend that the VLAN of the vPC keepalive SVI be allowed on only the interconnecting dedicated link and disallowed on all other links, including the peer link.
- You cannot have a link for non-vPC traffic in parallel with a vPC topology. This can cause errors with the traffic forwarding logic resulting in duplicate or missed packets.
- You cannot configure non-vPC interfaces across host ports on two different FEXs.

# Configuring vPCs

## Enabling vPCs

You must enable the vPC feature before you can configure and use vPCs.

**Procedure**

	<b>Command or Action</b>	<b>Purpose</b>
<b>Step 1</b>	switch# <b>configure terminal</b>	Enters global configuration mode.
<b>Step 2</b>	switch(config)# <b>feature vpc</b>	Enables vPCs on the switch.
<b>Step 3</b>	switch# <b>show feature</b>	(Optional) Displays which features are enabled on the switch.
<b>Step 4</b>	switch# <b>copy running-config startup-config</b>	(Optional) Copies the running configuration to the startup configuration.

This example shows how to enable the vPC feature:

```
switch# configure terminal
switch(config)# feature vpc
```

**Disabling vPCs**

You can disable the vPC feature.




---

**Note** When you disable the vPC feature, the Cisco Nexus device clears all the vPC configurations.

---

**Procedure**

	<b>Command or Action</b>	<b>Purpose</b>
<b>Step 1</b>	switch# <b>configure terminal</b>	Enters global configuration mode.
<b>Step 2</b>	switch(config)# <b>no feature vpc</b>	Disables vPCs on the switch.
<b>Step 3</b>	switch# <b>show feature</b>	(Optional) Displays which features are enabled on the switch.
<b>Step 4</b>	switch# <b>copy running-config startup-config</b>	(Optional) Copies the running configuration to the startup configuration.

This example shows how to disable the vPC feature:

```
switch# configure terminal
switch(config)# no feature vpc
```

## Creating a vPC Domain

You must create identical vPC domain IDs on both the vPC peer devices. This domain ID is used to automatically form the vPC system MAC address.

### Before You Begin

Ensure that you have enabled the vPC feature.

You must configure both switches on either side of the vPC peer link.

### Procedure

	<b>Command or Action</b>	<b>Purpose</b>
<b>Step 1</b>	switch# <b>configure terminal</b>	Enters global configuration mode.
<b>Step 2</b>	switch(config)# <b>vpc domain domain-id</b>	Creates a vPC domain on the switch, and enters the vpc-domain configuration mode. There is no default <i>domain-id</i> ; the range is from 1 to 1000.  <b>Note</b> You can also use the <b>vpc domain</b> command to enter the vpc-domain configuration mode for an existing vPC domain.
<b>Step 3</b>	switch# <b>show vpc brief</b>	(Optional) Displays brief information about each vPC domain.
<b>Step 4</b>	switch# <b>copy running-config startup-config</b>	(Optional) Copies the running configuration to the startup configuration.

This example shows how to create a vPC domain:

```
switch# configure terminal
switch(config) # vpc domain 5
```

## Configuring a vPC Keepalive Link and Messages

You can configure the destination IP for the peer-keepalive link that carries the keepalive messages. Optionally, you can configure other parameters for the keepalive messages.

The Cisco NX-OS software uses the peer-keepalive link between the vPC peers to transmit periodic, configurable keepalive messages. You must have Layer 3 connectivity between the peer devices to transmit these messages. The system cannot bring up the vPC peer link unless the peer-keepalive link is already up and running.

Ensure that both the source and destination IP addresses used for the peer-keepalive message are unique in your network and these IP addresses are reachable from the Virtual Routing and Forwarding (VRF) instance associated with the vPC peer-keepalive link.

**Note**

We recommend that you configure a separate VRF instance and put a Layer 3 port from each vPC peer switch into that VRF instance for the vPC peer-keepalive link. Do not use the peer link itself to send vPC peer-keepalive messages. For information on creating and configuring VRFs, see the Unicast Routing Configuration Guide for your device.

**Before You Begin**

Ensure that you have enabled the vPC feature.

You must configure the vPC peer-keepalive link before the system can form the vPC peer link.

You must configure both switches on either side of the vPC peer link.

**Procedure**

	<b>Command or Action</b>	<b>Purpose</b>
<b>Step 1</b>	switch# <b>configure terminal</b>	Enters global configuration mode.
<b>Step 2</b>	switch(config)# <b>vpc domain domain-id</b>	Creates a vPC domain on the switch if it does not already exist, and enters the vpc-domain configuration mode.
<b>Step 3</b>	switch(config-vpc-domain)# <b>peer-keepalive destination ipaddress [hold-timeout secs   interval msecs {timeout secs}   precedence {prec-value   network   internet   critical   flash-override   flash   immediate priority   routine}   tos {tos-value   max-reliability   max-throughput   min-delay   min-monetary-cost   normal}   tos-byte tos-byte-value}   source ipaddress   vrf {name   management vpc-keepalive}]</b>	Configures the IPv4 address for the remote end of the vPC peer-keepalive link.  <b>Note</b> The system does not form the vPC peer link until you configure a vPC peer-keepalive link. The management ports and VRF are the defaults.
<b>Step 4</b>	switch(config-vpc-domain)# <b>vpc peer-keepalive destination ipaddress source ipaddress</b>	(Optional) Configures a separate VRF instance and puts a Layer 3 port from each vPC peer device into that VRF for the vPC peer-keepalive link.
<b>Step 5</b>	switch# <b>show vpc peer-keepalive</b>	(Optional) Displays information about the configuration for the keepalive messages.
<b>Step 6</b>	switch# <b>copy running-config startup-config</b>	(Optional) Copies the running configuration to the startup configuration.

This example shows how to configure the destination IP address for the vPC-peer-keepalive link:

```
switch# configure terminal
```

```
switch(config)# vpc domain 5
switch(config-vpc-domain)# peer-keepalive destination 10.10.10.42
```

This example shows how to set up the peer keepalive link connection between the primary and secondary vPC device:

```
switch(config)# vpc domain 100
switch(config-vpc-domain)# peer-keepalive destination 192.168.2.2 source 192.168.2.1
Note:-----: Management VRF will be used as the default VRF :-----
switch(config-vpc-domain) #
```

This example shows how to create a separate VRF named vpc\_keepalive for the vPC keepalive link and how to verify the new VRF:

```
vrf context vpc_keepalive
interface Ethernet1/31
  switchport access vlan 123
interface Vlan123
  vrf member vpc_keepalive
  ip address 123.1.1.2/30
  no shutdown
vpc domain 1
  peer-keepalive destination 123.1.1.1 source 123.1.1.2 vrf
vpc_keepalive

L3-NEXUS-2# show vpc peer-keepalive

vPC keep-alive status          : peer is alive
--Peer is alive for           : (154477) seconds, (908) msec
--Send status                 : Success
--Last send at                : 2011.01.14 19:02:50 100 ms
--Sent on interface           : Vlan123
--Receive status              : Success
--Last receive at             : 2011.01.14 19:02:50 103 ms
--Received on interface       : Vlan123
--Last update from peer       : (0) seconds, (524) msec

vPC Keep-alive parameters
--Destination                 : 123.1.1.1
--Keepalive interval          : 1000 msec
--Keepalive timeout           : 5 seconds
--Keepalive hold timeout      : 3 seconds
--Keepalive vrf                : vpc_keepalive
--Keepalive udp port          : 3200
--Keepalive tos                : 192

The services provided by the switch , such as ping, ssh, telnet,
radius, are VRF aware. The VRF name need to be configured or
specified in order for the correct routing table to be used.
L3-NEXUS-2# ping 123.1.1.1 vrf vpc_keepalive
PING 123.1.1.1 (123.1.1.1): 56 data bytes
64 bytes from 123.1.1.1: icmp_seq=0 ttl=254 time=3.234 ms
64 bytes from 123.1.1.1: icmp_seq=1 ttl=254 time=4.931 ms
64 bytes from 123.1.1.1: icmp_seq=2 ttl=254 time=4.965 ms
64 bytes from 123.1.1.1: icmp_seq=3 ttl=254 time=4.971 ms
64 bytes from 123.1.1.1: icmp_seq=4 ttl=254 time=4.915 ms

--- 123.1.1.1 ping statistics ---
5 packets transmitted, 5 packets received, 0.00% packet loss
round-trip min/avg/max = 3.234/4.603/4.971 ms
```

## Creating a vPC Peer Link

You can create a vPC peer link by designating the EtherChannel that you want on each switch as the peer link for the specified vPC domain. We recommend that you configure the EtherChannels that you are designating

as the vPC peer link in trunk mode and that you use two ports on separate modules on each vPC peer switch for redundancy.

### Before You Begin

Ensure that you have enabled the vPC feature.

You must configure both switches on either side of the vPC peer link

### Procedure

	<b>Command or Action</b>	<b>Purpose</b>
<b>Step 1</b>	switch# <b>configure terminal</b>	Enters global configuration mode.
<b>Step 2</b>	switch(config)# <b>interface port-channel</b> <i>channel-number</i>	Selects the EtherChannel that you want to use as the vPC peer link for this switch, and enters the interface configuration mode.
<b>Step 3</b>	switch(config-if)# <b>vpc peer-link</b>	Configures the selected EtherChannel as the vPC peer link, and enters the vpc-domain configuration mode.
<b>Step 4</b>	switch# <b>show vpc brief</b>	(Optional) Displays information about each vPC, including information about the vPC peer link.
<b>Step 5</b>	switch# <b>copy running-config startup-config</b>	(Optional) Copies the running configuration to the startup configuration.

This example shows how to configure a vPC peer link:

```
switch# configure terminal
switch(config)# interface port-channel 20
switch(config-if)# vpc peer-link
```

### Checking the Configuration Compatibility

After you have configured the vPC peer link on both vPC peer switches, check that the configurations are consistent on all vPC interfaces.



#### Note

Beginning with Cisco NX-OS Release 5.0(2)N1(1), the following QoS parameters support Type 2 consistency checks:

- Network QoS—MTU and Pause
- Input Queuing —Bandwidth and Absolute Priority
- Output Queuing—Bandwidth and Absolute Priority

---

In the case of a Type 2 mismatch, the vPC is not suspended. Type 1 mismatches suspend the vPC.

Command or Action	Purpose
<b>switch# show vpc consistency-parameters {global   interface port-channel <i>channel-number</i>}</b>	Displays the status of those parameters that must be consistent across all vPC interfaces.

This example shows how to check that the required configurations are compatible across all the vPC interfaces:

```
switch# show vpc consistency-parameters global
Legend:
      Type 1 : vPC will be suspended in case of mismatch
Name          Type  Local Value          Peer Value
-----
QoS           2     ([], [], [], [], [],      ([], [], [], [], [], [])
              []))

Network QoS (MTU)    2     (1538, 0, 0, 0, 0, 0)  (1538, 0, 0, 0, 0, 0)
Network Qos (Pause)  2     (F, F, F, F, F, F)      (1538, 0, 0, 0, 0, 0)
Input Queuing (Bandwidth) 2     (100, 0, 0, 0, 0, 0)  (100, 0, 0, 0, 0, 0)
Input Queuing (Absolute Priority) 2     (F, F, F, F, F, F)  (100, 0, 0, 0, 0, 0)
Output Queuing (Bandwidth) 2     (100, 0, 0, 0, 0, 0)  (100, 0, 0, 0, 0, 0)
Output Queuing (Absolute Priority) 2     (F, F, F, F, F, F)  (100, 0, 0, 0, 0, 0)
STP Mode           1     Rapid-PVST          Rapid-PVST
STP Disabled        1     None               None
STP MST Region Name 1     ""                 ""
STP MST Region Revision 1     0                  0
STP MST Region Instance to VLAN Mapping 1

STP Loopguard       1     Disabled            Disabled
STP Bridge Assurance 1     Enabled             Enabled
STP Port Type, Edge 1     Normal, Disabled, Disabled
BPDUFilter, Edge BPDUGuard 1     Disabled            Disabled
STP MST Simulate PVST 1     Enabled            Enabled
Allowed VLANs        -     1,624              1
Local suspended VLANs -     624                -
switch#
```

This example shows how to check that the required configurations are compatible for an EtherChannel interface:

```
switch# show vpc consistency-parameters interface port-channel 20
Legend:
      Type 1 : vPC will be suspended in case of mismatch
Name          Type  Local Value          Peer Value
-----
Fex id         1     20                20
STP Port Type   1     Default           Default
STP Port Guard  1     None              None
STP MST Simulate PVST 1     Default           Default
mode           1     on                on
Speed          1     10 Gb/s          10 Gb/s
Duplex          1     full              full
Port Mode        1     fex-fabric       fex-fabric
Shut Lan         1     No                No
Allowed VLANs    -     1,3-3967,4048-4093  1-3967,4048-4093
```

## Enabling vPC Auto-Recovery

### Procedure

	Command or Action	Purpose
<b>Step 1</b>	switch# <b>configure terminal</b>	Enters global configuration mode.
<b>Step 2</b>	switch(config)# <b>vpc domain domain-id</b>	Enters vpc-domain configuration mode for an existing vPC domain.
<b>Step 3</b>	switch(config-vpc-domain)# <b>auto-recovery reload-delay delay</b>	Enables the auto-recovery feature and sets the reload delay period. The default is disabled.

This example shows how to enable the auto-recovery feature in vPC domain 10 and set the delay period for 240 seconds:

```
switch(config)# vpc domain 10
switch(config-vpc-domain)# auto-recovery reload-delay 240
Warning:
    Enables restoring of vPCs in a peer-detached state after reload, will wait for 240 seconds
    (by default) to determine if peer is un-reachable
```

This example shows how to view the status of the auto-recovery feature in vPC domain 10:

```
switch(config-vpc-domain)# show running-config vpc
!Command: show running-config vpc
!Time: Tue Dec  7 02:38:44 2010

version 5.0(2)N2(1)

feature vpc
vpc domain 10
  peer-keepalive destination 10.193.51.170
  auto-recovery
```

## Suspending Orphan Ports on a Secondary Switch in a vPC Topology

You can suspend a non-virtual port channel (vPC) port when a vPC secondary peer link goes down. A non-vPC port or an orphan port is a port that is not part of a vPC.



### Note

When a port is configured as an orphan port, the port will flap. This occurs because the system reevaluates whether the port can be brought up, given the constraints of the orphan port. For example, a Multichassis Etherchannel Trunk (MCT) needs to be up and election needs to be complete.

### Before You Begin

Enable the vPC feature.

## Procedure

	Command or Action	Purpose
<b>Step 1</b>	switch# <b>configure terminal</b>	Enters global configuration mode.
<b>Step 2</b>	switch(config)# <b>interface ethernet slot/port</b>	Specifies the port that you want to configure and enters interface configuration mode.
<b>Step 3</b>	switch(config-if)# <b>vpc orphan-port suspend</b>	Suspends the specified port if the secondary switch goes down.  <b>Note</b> The <b>vpc-orphan-port suspend</b> command is supported only on physical ports, and not supported on port-channel member ports. <b>Note</b> From Cisco NX-OS Release 7.3(1)N1(1), <b>vpc-orphan-port suspend</b> command is supported on Layer 3 ports too.
<b>Step 4</b>	switch(config-if)# <b>exit</b>	Exits interface configuration mode.
<b>Step 5</b>	switch# <b>show vpc orphan-port</b>	(Optional) Displays the orphan port configuration.
<b>Step 6</b>	switch(config-if)# <b>copy running-config startup-config</b>	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

This example shows how to suspend an orphan port:

```
switch# configure terminal
switch(config)# interface ethernet 1/20
switch(config-if)# vpc orphan-port suspend
```

This example shows how to display ports that are not part of the vPC but that share common VLANs with ports that are part of the vPC:

```
switch# configure terminal
switch(config)# show vpc orphan-ports
Note:
-----::Going through port database. Please be patient.::-----
VLAN Orphan Ports
-----
1 Po600
2 Po600
3 Po600
4 Po600
5 Po600
6 Po600
7 Po600
8 Po600
9 Po600
10 Po600
11 Po600
12 Po600
13 Po600
14 Po600
...
```

## Creating an EtherChannel Host Interface

To connect to a downstream server from a Cisco Nexus Fabric Extender you can create a EtherChannel host interface. An EtherChannel host interface can have only one host interface as a member depending on the fabric extender model. The Cisco Nexus 2148T allows only one interface member per fabric extender, newer fabric extenders allow up to 8 members of the same port-channel on a single fabric extender. You need to create an EtherChannel host interface to configure a vPC on it that uses the Fabric Extender topology.

### Before You Begin

Ensure that you have enabled the vPC feature.

Ensure that the connected Fabric Extender is online.

You must configure both switches on either side of the vPC peer link with the following procedure.

### Procedure

	<b>Command or Action</b>	<b>Purpose</b>
<b>Step 1</b>	switch# <b>configure terminal</b>	Enters global configuration mode.
<b>Step 2</b>	switch(config)# <b>interface ethernet chassis/slot/port</b>	Specifies an interface to configure, and enters interface configuration mode.
<b>Step 3</b>	switch(config-if)# <b>channel-group channel-number mode {active   passive   on}</b>	Creates an EtherChannel host interface on the selected host interface.
<b>Step 4</b>	switch(config-if)# <b>show port-channel summary</b>	(Optional) Displays information about each EtherChannel host interface.
<b>Step 5</b>	switch(config-if)# <b>copy running-config startup-config</b>	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

This example shows how to configure an EtherChannel host interface:

```
switch# configure terminal
switch(config)# interface ethernet 101/1/20
switch(config-if)# channel-group 7 mode active
```

## Moving Other Port Channels into a vPC

### Before You Begin

Ensure that you have enabled the vPC feature.

You must configure both switches on either side of the vPC peer link with the following procedure.

## Procedure

	<b>Command or Action</b>	<b>Purpose</b>
<b>Step 1</b>	switch# <b>configure terminal</b>	Enters global configuration mode.
<b>Step 2</b>	switch(config)# <b>interface port-channel channel-number</b>	Selects the port channel that you want to put into the vPC to connect to the downstream switch, and enters interface configuration mode.  <b>Note</b> A vPC can be configured on a normal port channel (physical vPC topology), on an port channel fabric interface (fabric extender vPC topology), and on a port channel host interface (host interface vPC topology)
<b>Step 3</b>	switch(config-if)# <b>vpc number</b>	Configures the selected port channel into the vPC to connect to the downstream switch. The range is from 1 to 4096.  The vPC <i>number</i> that you assign to the port channel that connects to the downstream switch from the vPC peer switch must be identical on both vPC peer switches.
<b>Step 4</b>	switch# <b>show vpc brief</b>	(Optional) Displays information about each vPC.
<b>Step 5</b>	switch# <b>copy running-config startup-config</b>	(Optional) Copies the running configuration to the startup configuration.

This example shows how to configure a port channel that will connect to the downstream device:

```
switch# configure terminal
switch(config)# interface port-channel 20
switch(config-if)# vpc 5
```

## Manually Configuring a vPC Domain MAC Address



**Note** Configuring the system address is an optional configuration step.

### Before You Begin

Ensure that you have enabled the vPC feature.

You must configure both switches on either side of the vPC peer link.

## Procedure

	<b>Command or Action</b>	<b>Purpose</b>
<b>Step 1</b>	switch# <b>configure terminal</b>	Enters global configuration mode.

## Manually Configuring the System Priority

	<b>Command or Action</b>	<b>Purpose</b>
<b>Step 2</b>	switch(config)# vpc domain <i>domain-id</i>	Selects an existing vPC domain on the switch, or creates a new vPC domain, and enters the vpc-domain configuration mode. There is no default <i>domain-id</i> ; the range is from 1 to 1000.
<b>Step 3</b>	switch(config-vpc-domain)# system-mac <i>mac-address</i>	Enters the MAC address that you want for the specified vPC domain in the following format: aaaa.bbbb.cccc.
<b>Step 4</b>	switch# show vpc role	(Optional) Displays the vPC system MAC address.
<b>Step 5</b>	switch# copy running-config startup-config	(Optional) Copies the running configuration to the startup configuration.

This example shows how to configure a vPC domain MAC address:

```
switch# configure terminal
switch(config)# vpc domain 5
switch(config-if)# system-mac 23fb.4ab5.4c4e
```

## Manually Configuring the System Priority

When you create a vPC domain, the system automatically creates a vPC system priority. However, you can also manually configure a system priority for the vPC domain.

### Before You Begin

Ensure that you have enabled the vPC feature.

You must configure both switches on either side of the vPC peer link.

### Procedure

	<b>Command or Action</b>	<b>Purpose</b>
<b>Step 1</b>	switch# configure terminal	Enters global configuration mode.
<b>Step 2</b>	switch(config)# vpc domain <i>domain-id</i>	Selects an existing vPC domain on the switch, or creates a new vPC domain, and enters the vpc-domain configuration mode. There is no default <i>domain-id</i> ; the range is from 1 to 1000.
<b>Step 3</b>	switch(config-vpc-domain)# system-priority <i>priority</i>	Enters the system priority that you want for the specified vPC domain. The range of values is from 1 to 65535. The default value is 32667.

	<b>Command or Action</b>	<b>Purpose</b>
<b>Step 4</b>	switch# <b>show vpc brief</b>	(Optional) Displays information about each vPC, including information about the vPC peer link.
<b>Step 5</b>	switch# <b>copy running-config startup-config</b>	(Optional) Copies the running configuration to the startup configuration.

This example shows how to configure a vPC peer link:

```
switch# configure terminal
switch(config)# vpc domain 5
switch(config-if)# system-priority 4000
```

## Manually Configuring a vPC Peer Switch Role

By default, the Cisco NX-OS software elects a primary and secondary vPC peer switch after you configure the vPC domain and both sides of the vPC peer link. However, you may want to elect a specific vPC peer switch as the primary switch for the vPC. Then, you would manually configure the role value for the vPC peer switch that you want as the primary switch to be lower than the other vPC peer switch.

vPC does not support role preemption. If the primary vPC peer switch fails, the secondary vPC peer switch takes over to become operationally the vPC primary switch. However, the original operational roles are not restored when the formerly primary vPC comes up again.

### Before You Begin

Ensure that you have enabled the vPC feature.

You must configure both switches on either side of the vPC peer link.

### Procedure

	<b>Command or Action</b>	<b>Purpose</b>
<b>Step 1</b>	switch# <b>configure terminal</b>	Enters global configuration mode.
<b>Step 2</b>	switch(config)# <b>vpc domain domain-id</b>	Selects an existing vPC domain on the switch, or creates a new vPC domain, and enters the vpc-domain configuration mode. There is no default <i>domain-id</i> ; the range is from 1 to 1000.
<b>Step 3</b>	switch(config-vpc-domain)# <b>role priority priority</b>	Enters the role priority that you want for the vPC system priority. The range of values is from 1 to 65535. The default value is 32667.

	Command or Action	Purpose
<b>Step 4</b>	switch# <b>show vpc brief</b>	(Optional) Displays information about each vPC, including information about the vPC peer link.
<b>Step 5</b>	switch# <b>copy running-config startup-config</b>	(Optional) Copies the running configuration to the startup configuration.

This example shows how to configure a vPC peer link:

```
switch# configure terminal
switch(config)# vpc domain 5
switch(config-if)# role priority 4000
```

## Configuring the vPC Peer Switch

### Configuring a Pure vPC Peer Switch Topology

You can configure a pure vPC peer switch topology using the **peer-switch** command and then you set the best possible (lowest) spanning tree bridge priority value.



**Note**

The values you apply for the spanning tree priority must be identical on both vPC peers.

#### Before You Begin

Ensure that you have enabled the vPC feature.

#### Procedure

	Command or Action	Purpose
<b>Step 1</b>	switch# <b>configure terminal</b>	Enters global configuration mode.
<b>Step 2</b>	switch(config)# <b>vpc domain domain-id</b>	Enters the vPC domain number that you want to configure. The system enters the vpc-domain configuration mode.
<b>Step 3</b>	switch(config-vpc-domain)# <b>peer-switch</b>	Enables the vPC switch pair to appear as a single STP root in the Layer 2 topology. Use the <b>no</b> form of the command to disable the peer switch vPC topology.
<b>Step 4</b>	switch(config-vpc-domain)# <b>spanning-tree vlan vlan-range priority value</b>	Configures the bridge priority of the VLAN. Valid values are multiples of 4096. The default value is 32768.

	<b>Command or Action</b>	<b>Purpose</b>
		<b>Note</b> This value must be identical on both vPC peers.
<b>Step 5</b>	switch(config-vpn-domain)# <b>exit</b>	Exits the vpc-domain configuration mode.
<b>Step 6</b>	switch(config)# <b>show spanning-tree summary</b>	(Optional) Displays a summary of the spanning tree port states including the vPC peer switch.  Look for the following line in the command output: vPC peer-switch is enabled (operational)
<b>Step 7</b>	switch(config)# <b>copy running-config startup-config</b>	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

This example shows how to configure a pure vPC peer switch topology:

```

switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# vpc domain 5
switch(config-vpc-domain)# peer-switch
2010 Apr 28 14:44:44 switch %STP-2-VPC_PEER_SWITCH_CONFIG_ENABLED: vPC peer-switch
configuration is enabled. Please make sure to configure spanning tree "bridge" priority as
per recommended guidelines to make vPC peer-switch operational.
switch(config-vpc-domain)# exit
switch(config)# spanning-tree vlan 1 priority 8192
switch(config)# show spanning-tree summary
Switch is in rapid-pvst mode
Root bridge for: VLAN0001-VLAN0050, VLAN0100-VLAN0149, VLAN0200-VLAN0249
    VLAN0300-VLAN0349, VLAN0400-VLAN0599, VLAN0900-VLAN0999
Port Type Default           is disable
Edge Port [PortFast] BPDU Guard Default is disabled
Edge Port [PortFast] BPDU Filter Default is disabled
Bridge Assurance            is enabled
Loopguard Default          is disabled
Pathcost method used       is short
vPC peer-switch             is enabled (operational)
Name          Blocking Listening Learning Forwarding STP Active
----- -----
VLAN0001      0          0          0          16         16
VLAN0002      0          0          0          16         16
switch(config)# copy running-config startup-config
switch(config)#

```

## Configuring a Hybrid vPC Peer Switch Topology

You can configure a hybrid vPC and non-vPC peer switch topology by using the spanning-tree pseudo-information command to change the designated bridge ID so that it meets the STP VLAN-based load-balancing criteria and then change the root bridge ID priority to a value that is better than the best bridge priority. You then enable the peer switch. For more information, see the command reference for your device.

**Note**

If you previously configured global spanning tree parameters and you subsequently configure spanning tree pseudo information parameters, be aware that the pseudo information parameters take precedence over the global parameters.

**Before You Begin**

Ensure that you have enabled the vPC feature.

**Procedure**

	<b>Command or Action</b>	<b>Purpose</b>
<b>Step 1</b>	switch# <b>configure terminal</b>	Enters global configuration mode.
<b>Step 2</b>	switch(config)# <b>spanning-tree pseudo-information</b>	Configures the spanning tree pseudo information. <b>Note</b> This configuration takes precedence over any global spanning tree configurations.
<b>Step 3</b>	switch(config-pseudo)# <b>vlan vlan-id designated priority priority</b>	Configures the designated bridge priority of the VLAN. Valid values are multiples of 4096 from 0 to 61440.
<b>Step 4</b>	switch(config-pseudo)# <b>vlan vlan-id root priority priority</b>	Configures the root bridge priority of the VLAN. Valid values are multiples of 4096 from 0 to 61440. <b>Note</b> This value must be identical on both vPC peers to have an operational peer switch.
<b>Step 5</b>	switch(config-pseudo)# <b>exit</b>	Exits spanning tree pseudo information configuration mode.
<b>Step 6</b>	switch(config)# <b>vpc domain domain-id</b>	Enters the vPC domain number that you want to configure. The system enters the vpc-domain configuration mode.
<b>Step 7</b>	switch(config-vpc-domain)# <b>peer-switch</b>	Enables the vPC switch pair to appear as a single STP root in the Layer 2 topology. Use the <b>no</b> form of the command to disable the peer switch vPC topology.
<b>Step 8</b>	switch(config-vpc-domain)# <b>exit</b>	Exits the vpc-domain configuration mode.
<b>Step 9</b>	switch(config)# <b>show spanning-tree summary</b>	(Optional) Displays a summary of the spanning tree port states including the vPC peer switch. Look for the following line in the command output: vPC peer-switch is enabled (operational)
<b>Step 10</b>	switch(config)# <b>copy running-config startup-config</b>	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

This example shows how to configure a hybrid vPC peer switch topology:

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# spanning-tree pseudo-information
switch(config-pseudo)# vlan 1 designated priority 8192
switch(config-pseudo)# vlan 1 root priority 4096
switch(config-pseudo)# exit
switch(config)# vpc domain 5
switch(config-vpc-domain)# peer-switch
switch(config-vpc-domain)# exit
switch(config)# copy running-config startup-config
```

## Verifying the vPC Configuration

Use the following commands to display vPC configuration information:

Command	Purpose
switch# <b>show feature</b>	Displays whether vPC is enabled or not.
switch# <b>show port-channel capacity</b>	Displays how many EtherChannels are configured and how many are still available on the switch.
switch# <b>show running-config vpc</b>	Displays running configuration information for vPCs.
switch# <b>show vpc brief</b>	Displays brief information on the vPCs.
switch# <b>show vpc consistency-parameters</b>	Displays the status of those parameters that must be consistent across all vPC interfaces.
switch# <b>show vpc peer-keepalive</b>	Displays information on the peer-keepalive messages.
switch# <b>show vpc role</b>	Displays the peer status, the role of the local switch, the vPC system MAC address and system priority, and the MAC address and priority for the local vPC switch.
switch# <b>show vpc statistics</b>	Displays statistics on the vPCs. <b>Note</b> This command displays the vPC statistics only for the vPC peer device that you are working on.

For information about the switch output, see the Command Reference for your Cisco Nexus Series switch.

## Viewing the Graceful Type-1 Check Status

This example shows how to display the current status of the graceful Type-1 consistency check:

```
switch# show vpc brief
Legend:
(*) - local vPC is down, forwarding via vPC peer-link
```

## Viewing a Global Type-1 Inconsistency

```
vPC domain id : 10
Peer status : peer adjacency formed ok
vPC keep-alive status : peer is alive
Configuration consistency status: success
Per-vlan consistency status : success
Type-2 consistency status : success
vPC role : secondary
Number of vPCs configured : 34
Peer Gateway : Disabled
Dual-active excluded VLANs : -
Graceful Consistency Check : Enabled

vPC Peer-link status
-----
id Port Status Active vlans
-- -- --
1 Po1 up 1
```

## Viewing a Global Type-1 Inconsistency

When a global Type-1 inconsistency occurs, the vPCs on the secondary switch are brought down. The following example shows this type of inconsistency when there is a spanning-tree mode mismatch.

The example shows how to display the status of the suspended vPC VLANs on the secondary switch:

```
switch(config)# show vpc
Legend:
(*) - local vPC is down, forwarding via vPC peer-link

vPC domain id : 10
Peer status : peer adjacency formed ok
vPC keep-alive status : peer is alive
Configuration consistency status: failed
Per-vlan consistency status : success
Configuration consistency reason: vPC type-1 configuration incompatible - STP
Mode inconsistent
Type-2 consistency status : success
vPC role : secondary
Number of vPCs configured : 2
Peer Gateway : Disabled
Dual-active excluded VLANs : -
Graceful Consistency Check : Enabled

vPC Peer-link status
-----
id Port Status Active vlans
-- -- --
1 Po1 up 1-10

vPC status
-----
id Port Status Consistency Reason Active vlans
-- -- --
20 Po20 down* failed Global compat check failed -
30 Po30 down* failed Global compat check failed -
```

The example shows how to display the inconsistent status (the VLANs on the primary vPC are not suspended) on the primary switch:

```
switch(config)# show vpc
Legend:
(*) - local vPC is down, forwarding via vPC peer-link

vPC domain id : 10
Peer status : peer adjacency formed ok
vPC keep-alive status : peer is alive
Configuration consistency status: failed
Per-vlan consistency status : success
```

```

Configuration consistency reason: vPC type-1 configuration incompatible - STP Mo
de inconsistent
Type-2 consistency status      : success
vPC role                      : primary
Number of vPCs configured     : 2
Peer Gateway                  : Disabled
Dual-active excluded VLANs    : -
Graceful Consistency Check   : Enabled

vPC Peer-link status
-----
id  Port  Status Active vlans
--  --   --   --
1   Po1   up    1-10

vPC status
-----
id  Port  Status Consistency Reason          Active vlans
--  --   --   --
20  Po20  up    failed   Global compat check failed 1-10
30  Po30  up    failed   Global compat check failed 1-10

```

## Viewing an Interface-Specific Type-1 Inconsistency

When an interface-specific Type-1 inconsistency occurs, the vPC port on the secondary switch is brought down while the primary switch vPC ports remain up. The following example shows this type of inconsistency when there is a switchport mode mismatch.

This example shows how to display the status of the suspended vPC VLAN on the secondary switch:

```

switch(config-if)# show vpc brief
Legend:
(*) - local vPC is down, forwarding via vPC peer-link

vPC domain id          : 10
Peer status             : peer adjacency formed ok
vPC keep-alive status  : peer is alive
Configuration consistency status: success
Per-vlan consistency status : success
Type-2 consistency status : success
vPC role                : secondary
Number of vPCs configured : 2
Peer Gateway            : Disabled
Dual-active excluded VLANs : -
Graceful Consistency Check : Enabled

vPC Peer-link status
-----
id  Port  Status Active vlans
--  --   --   --
1   Po1   up    1

vPC status
-----
id  Port  Status Consistency Reason          Active vlans
--  --   --   --
20  Po20  up    success   success           1
30  Po30  down*  failed   Compatibility check failed -
                           for port mode

```

This example shows how to display the inconsistent status (the VLANs on the primary vPC are not suspended) on the primary switch:

```

switch(config-if)# show vpc brief
Legend:
(*) - local vPC is down, forwarding via vPC peer-link

```

## Viewing a Per-VLAN Consistency Status

```

vPC domain id          : 10
Peer status             : peer adjacency formed ok
vPC keep-alive status  : peer is alive
Configuration consistency status: success
Per-vlan consistency status : success
Type-2 consistency status : success
vPC role                : primary
Number of vPCs configured : 2
Peer Gateway            : Disabled
Dual-active excluded VLANs : -
Graceful Consistency Check : Enabled

vPC Peer-link status
-----
id  Port   Status Active vlans
--  ---   --  --
1   Po1    up      1

vPC status
-----
id  Port       Status Consistency Reason           Active vlans
--  ---       --  --  --
20  Po20      up     success      success          1
30  Po30      up     failed      Compatiblity check failed 1
                                         for port mode

```

## Viewing a Per-VLAN Consistency Status

To view the per-VLAN consistency or inconsistency status, enter the **show vpc consistency-parameters vlans** command.

This example shows how to display the consistent status of the VLANs on the primary and the secondary switches.

```

switch(config-if)# show vpc brief
Legend:
(*) - local vPC is down, forwarding via vPC peer-link

vPC domain id          : 10
Peer status             : peer adjacency formed ok
vPC keep-alive status  : peer is alive
Configuration consistency status: success
Per-vlan consistency status : success
Type-2 consistency status : success
vPC role                : secondary
Number of vPCs configured : 2
Peer Gateway            : Disabled
Dual-active excluded VLANs : -
Graceful Consistency Check : Enabled

vPC Peer-link status
-----
id  Port   Status Active vlans
--  ---   --  --
1   Po1    up      1-10

vPC status
-----
id  Port       Status Consistency Reason           Active vlans
--  ---       --  --  --
20  Po20      up     success      success          1-10
30  Po30      up     success      success          1-10

```

Entering **no spanning-tree vlan 5** command triggers the inconsistency on the primary and secondary VLANs:

```
switch(config)# no spanning-tree vlan 5
```

This example shows how to display the per-VLAN consistency status as Failed on the secondary switch:

```
switch(config)# show vpc brief
Legend:
(*) - local vPC is down, forwarding via vPC peer-link

vPC domain id : 10
Peer status : peer adjacency formed ok
vPC keep-alive status : peer is alive
Configuration consistency status: success
Per-vlan consistency status : failed
Type-2 consistency status : success
vPC role : secondary
Number of vPCs configured : 2
Peer Gateway : Disabled
Dual-active excluded VLANs : -
Graceful Consistency Check : Enabled

vPC Peer-link status
-----
id Port Status Active vlans
-- --
1 Po1 up 1-4,6-10

vPC status
-----
id Port Status Consistency Reason Active vlans
-- --
20 Po20 up success success 1-4,6-10
30 Po30 up success success 1-4,6-10
```

This example shows how to display the per-VLAN consistency status as Failed on the primary switch:

```
switch(config)# show vpc brief
Legend:
(*) - local vPC is down, forwarding via vPC peer-link

vPC domain id : 10
Peer status : peer adjacency formed ok
vPC keep-alive status : peer is alive
Configuration consistency status: success
Per-vlan consistency status : failed
Type-2 consistency status : success
vPC role : primary
Number of vPCs configured : 2
Peer Gateway : Disabled
Dual-active excluded VLANs : -
Graceful Consistency Check : Enabled

vPC Peer-link status
-----
id Port Status Active vlans
-- --
1 Po1 up 1-4,6-10

vPC status
-----
id Port Status Consistency Reason Active vlans
-- --
20 Po20 up success success 1-4,6-10
30 Po30 up success success 1-4,6-10
```

This example shows the inconsistency as STP Disabled:

```
switch(config)# show vpc consistency-parameters vlans

Name Type Reason Code Pass Vlans
----- -----
STP Mode 1 success 0-4095
STP Disabled 1 vPC type-1 configuration 0-4,6-4095
```

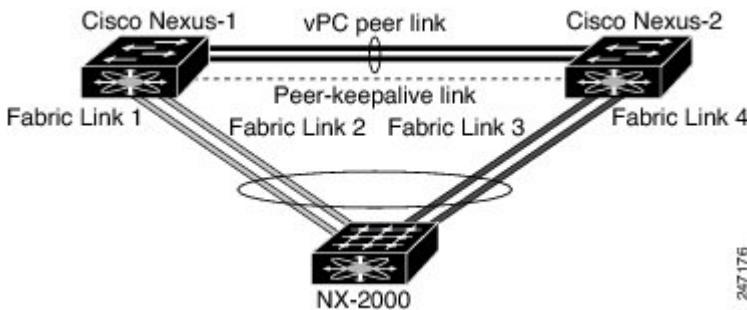
		<b>incompatible - STP is enabled or disabled on some or all vlans</b>	
STP MST Region Name	1	success	0-4095
STP MST Region Revision	1	success	0-4095
STP MST Region Instance to VLAN Mapping	1	success	0-4095
STP Loopguard	1	success	0-4095
STP Bridge Assurance	1	success	0-4095
STP Port Type, Edge BPDUFilter, Edge BPDUGuard	1	success	0-4095
STP MST Simulate PVST Pass Vlans	1	success	0-4095
	-		0-4, 6-4095

## vPC Example Configurations

### Dual Homed Fabric Extender vPC Configuration Example

The following example shows how to configure the dual homed Fabric Extender vPC topology using the management VRF to carry the peer-keepalive messages on switch CiscoNexus-1 as shown in following figure:

**Figure 4: vPC Configuration Example**



#### Before You Begin

Ensure that the Cisco Nexus 2000 Series Fabric Extender NX-2000-100 is attached and online.

#### Procedure

**Step 1** Enable vPC and LACP.

```
CiscoNexus-1# configure terminal
CiscoNexus-1(config)# feature lacp
CiscoNexus-1(config)# feature vpc
```

**Step 2** Create the vPC domain and add the vPC peer-keepalive link.

```
CiscoNexus-1(config)# vpc domain 1
CiscoNexus-1(config-vpc-domain)# peer-keepalive destination 10.10.10.237
CiscoNexus-1(config-vpc-domain)# exit
```

**Step 3** Configure the vPC peer link as a two port Etherchannel.

```
CiscoNexus-1(config)# interface ethernet 1/1-2
CiscoNexus-1(config-if-range)# switchport mode trunk
CiscoNexus-1(config-if-range)# switchport trunk allowed vlan 20-50
CiscoNexus-1(config-if-range)# switchport trunk native vlan 20
CiscoNexus-1(config-if-range)# channel-group 20 mode active
CiscoNexus-1(config-if-range)# exit
CiscoNexus-1(config)# interface port-channel 20
CiscoNexus-1(config-if)# vpc peer-link
CiscoNexus-1(config-if)# exit
```

**Step 4** Create a Fabric Extender identifier (for example, "100").

```
CiscoNexus-1(config)# fex 100
CiscoNexus-1(config-fex)# pinning max-links 1
CiscoNexus-1(fex)# exit
```

**Step 5** Configure the fabric EtherChannel links for the Fabric Extender 100.

```
CiscoNexus-1(config)# interface ethernet 1/20
CiscoNexus-1(config-if)# channel-group 100
CiscoNexus-1(config-if)# exit
CiscoNexus-1(config)# interface port-channel 100
CiscoNexus-1(config-if)# switchport mode fex-fabric
CiscoNexus-1(config-if)# vpc 100
CiscoNexus-1(config-if)# fex associate 100
CiscoNexus-1(config-if)# exit
```

**Step 6** Configure each host interface port on the Fabric Extender 100 on both Cisco Nexus devices as for all the other steps.

```
CiscoNexus-1(config)# interface ethernet 100/1/1-48
CiscoNexus-1(config-if)# switchport mode access
CiscoNexus-1(config-if)# switchport access vlan 50
CiscoNexus-1(config-if)# no shutdown
CiscoNexus-1(config-if)# exit
```

**Step 7** Save the configuration.

```
CiscoNexus-1(config)# copy running-config startup-config
```

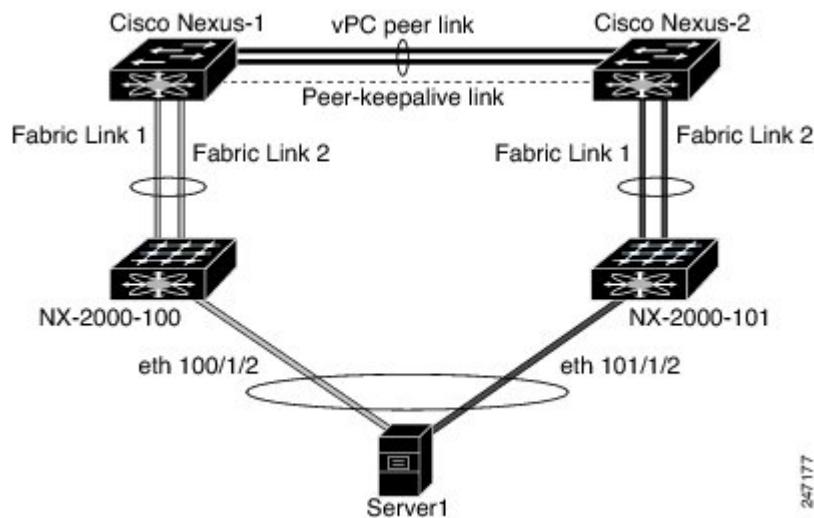
Repeat all the above steps for the CiscoNexus-2 switch.

---

## Single Homed Fabric Extender vPC Configuration Example

The following example shows how to configure the single homed Fabric Extender vPC topology using the default VRF to carry the peer-keepalive messages on switch CiscoNexus-1 as shown in following figure:

**Figure 5: vPC Configuration Example**



**Note** The following example only shows the configuration of CiscoNexus-1 which is connected to the Fabric Extender NX-2000-100. You must repeat these steps on its vPC peer, CiscoNexus-2, which is connected to the Fabric Extender NX-2000-101.

### Before You Begin

Ensure that the Cisco Nexus 2000 Series Fabric Extenders NX-2000-100 and NX-2000-101 are attached and online.

### Procedure

---

#### Step 1 Enable vPC and LACP.

```
CiscoNexus-1# configure terminal
CiscoNexus-1(config)# feature lacp
CiscoNexus-1(config)# feature vpc
```

- Step 2** Enable SVI interfaces, create the VLAN and SVI to be used by the vPC peer-keepalive link.

```
CiscoNexus-1(config)# feature interface-vlan
CiscoNexus-1(config)# vlan 900
CiscoNexus-1(config-vlan)# int vlan 900
CiscoNexus-1(config-if)# ip address 10.10.10.236 255.255.255.0
CiscoNexus-1(config-if)# no shutdown
CiscoNexus-1(config-if)# exit
```

- Step 3** Create the vPC domain and add the vPC peer-keepalive link in the default VRF.

```
CiscoNexus-1(config)# vpc domain 30
CiscoNexus-1(config-vpc-domain)# peer-keepalive destination 10.10.10.237 source 10.10.10.236
  vrf default
CiscoNexus-1(config-vpc-domain)# exit
```

**Note** VLAN 900 must **not** be trunked across the vPC peer-link because it carries the vPC peer-keepalive messages. There must be an alternative path between switches CiscoNexus-1 and CiscoNexus-2 for the vPC peer-keepalive messages.

- Step 4** Configure the vPC peer link as a two port Etherchannel.

```
CiscoNexus-1(config)# interface ethernet 1/1-2
CiscoNexus-1(config-if-range)# switchport mode trunk
CiscoNexus-1(config-if-range)# switchport trunk allowed vlan 20-50
CiscoNexus-1(config-if-range)# switchport trunk native vlan 20
CiscoNexus-1(config-if-range)# channel-group 30 mode active
CiscoNexus-1(config-if-range)# exit
CiscoNexus-1(config)# interface port-channel 30
CiscoNexus-1(config-if)# vpc peer-link
CiscoNexus-1(config-if)# exit
```

- Step 5** Configure the Fabric Extender NX-2000-100.

```
CiscoNexus-1(config)# fex 100
CiscoNexus-1(config-fex)# pinning max-links 1
CiscoNexus-1(fex)# exit
```

- Step 6** Configure the fabric EtherChannel links for the Fabric Extender NX-2000-100.

```
CiscoNexus-1(config)# interface ethernet 1/20-21
CiscoNexus-1(config-if)# channel-group 100
CiscoNexus-1(config-if)# exit
CiscoNexus-1(config)# interface port-channel 100
CiscoNexus-1(config-if)# switchport mode fex-fabric
CiscoNexus-1(config-if)# fex associate 100
CiscoNexus-1(config-if)# exit
```

**Step 7** Configure a vPC server port on the Fabric Extender NX-2000-100.

```
CiscoNexus-1(config-if)# interface ethernet 100/1/1
CiscoNexus-1(config-if)# switchport mode trunk
CiscoNexus-1(config-if)# switchport trunk native vlan 100
CiscoNexus-1(config-if)# switchport trunk allowed vlan 100-105
CiscoNexus-1(config-if)# channel-group 600
CiscoNexus-1(config-if)# no shutdown
CiscoNexus-1(config-if)# exit
CiscoNexus-1(config)# interface port-channel 600
CiscoNexus-1(config-if)# vpc 600
CiscoNexus-1(config-if)# no shutdown
CiscoNexus-1(config-if)# exit
```

**Step 8** Save the configuration.

```
CiscoNexus-1(config)# copy running-config startup-config
```

---

## vPC Default Settings

The following table lists the default settings for vPC parameters.

**Table 1: Default vPC Parameters**

Parameters	Default
vPC system priority	32667
vPC peer-keepalive message	Disabled
vPC peer-keepalive interval	1 second
vPC peer-keepalive timeout	5 seconds
vPC peer-keepalive UDP port	3200