Configuring FCoE VLANs and Virtual Interfaces

This chapter contains the following sections:

Information About Virtual Interfaces

Cisco Nexus devices support Fibre Channel over Ethernet (FCoE), which allows Fibre Channel and Ethernet traffic to be carried on the same physical Ethernet connection between the switch and the servers.

The Fibre Channel portion of FCoE is configured as a virtual Fibre Channel interface. Logical Fibre Channel features (such as interface mode) can be configured on virtual Fibre Channel interfaces.

A virtual Fibre Channel interface must be bound to an interface before it can be used. The binding is to a physical Ethernet interface (when the converged network adapter (CNA) is directly connected to the Cisco Nexus device), a MAC address (when the CNA is remotely connected over a Layer 2 bridge), or an EtherChannel when the CNA connects to the Fibre Channel Forwarder (FCF) over a virtual port channel (vPC).

VE Port

A virtual expansion (VE) port acts as an expansion port in an FCoE network. VE ports can connect multiple FCoE switches together in the network. You can bind a VE port to a physical ethernet port or a port channel.

On the Cisco Nexus 5000 and 6000 Series switches, traffic across members of a port channel that a VE_Port is bound to is load balanced based on SID, DID, and OXID.

In order to enable all links to be used in the port-channel for FCoE traffic, enter the **port-channel load-balance ethernet** *source-dest-port* command to configure 'port-channel load balancing' to 'source-dest-port'. The configuration 'source-destination-oxid' load balancing is used for FCoE traffic.

Guidelines and Limitations for FCoE VLANs and Virtual Interfaces

FCoE VLANs and Virtual Fiber Channel (vFC) interfaces have these guidelines and limitations:

- Each vFC interface must be bound to an FCoE-enabled Ethernet or EtherChannel interface or to the MAC address of a remotely connected adapter. FCoE is supported on 10-Gigabit Ethernet interfaces.

  The Ethernet or EtherChannel interface that you bind to the vFC interface must be configured as follows:

  - The Ethernet or EtherChannel interface must be a trunk port (use the **switchport mode trunk** command).

  - The FCoE VLAN that corresponds to a vFC's VSAN must be in the allowed VLAN list.

  - You must not configure an FCoE VLAN as the native VLAN of the trunk port.

  > **Note** The native VLAN is the default VLAN on a trunk. Any untagged frames transit the trunk as native VLAN traffic.

  - You should use an FCoE VLAN only for FCoE.

  - Do not use the default VLAN, VLAN1, as an FCoE VLAN.

  - You must configure the Ethernet interface as PortFast (use the **spanning-tree port type edge trunk** command).

  > **Note** You are not required to configure trunking on the server interface even if the switch interface is configured with trunking enabled. All non-FCoE traffic from the server is passed on the native VLAN.

- The vFC interface can be bound to Ethernet port channels with multiple member ports connected to FCoE Initialization Protocol (FIP) snooping bridges.

- Each vFC interface is associated with only one VSAN.

- You must map any VSAN with associated vFC interfaces to a dedicated FCoE-enabled VLAN.

- FCoE is not supported on private VLANs.

- If the converged access switches (in the same SAN fabric or in another) need to be connected to each other over Ethernet links for a LAN alternate path, then you must explicitly configure such links to exclude all FCoE VLANs from membership.

- You must use separate FCoE VLANs for FCoE in SAN-A and SAN-B fabrics.

- FCoE connectivity to pre-FIP CNAs over virtual port channels (vPCs) is not supported.

- The maximum number of vFCs that can be bound to a port-channel is 48 (24 for the Nexus 6001).

Note Virtual interfaces are created with the administrative state set to down. You must explicitly configure the administrative state to bring the virtual interface into operation.

Configuring Virtual Interfaces

Mapping a VSAN to a VLAN

A unique, dedicated VLAN must be configured at every converged access switch to carry traffic for each VSAN in the SAN (for example, VLAN 1002 for VSAN 1, VLAN 1003 for VSAN 2, and so on). If Multiple Spanning Tree (MST) is enabled, a separate MST instance must be used for FCoE VLANs.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **vlan** *vlan-id*	Enters VLAN configuration mode. The VLAN number range is from 1 to 4096.
Step 3	switch(config-vlan)# **fcoe** [**vsan** *vsan-id*]	Enables FCoE for the specified VLAN. If you do not specify a VSAN number, a mapping is created from this VLAN to the VSAN with the same number. Configures the mapping from this VLAN to the specified VSAN.
Step 4	switch(config-vlan)# **exit**	Exits VLAN configuration mode. You must exit this mode to execute the configured commands on your Cisco Nexus device.
Step 5	(Optional) switch(config)# **show vlan fcoe**	Displays information about the FCoE configuration for a VLAN.
Step 6	(Optional) switch(config-if)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to map VLAN 200 to VSAN 2:

```
switch(config)# vlan 200
switch(config-vlan)# fcoe vsan 2
```

Creating a Virtual Fibre Channel Interface

You can create a virtual Fibre Channel interface. You must bind the virtual Fibre Channel interface to a physical interface before it can be used.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **interface vfc** *vfc-id*	Creates a virtual Fibre Channel interface (if it does not already exist) and enters interface configuration mode.
		The virtual Fibre Channel interface ID range is from 1 to 8192.
Step 3	switch(config-if)# **bind** {**interface** {**ethernet** *slot*/*port* | **port-channel** *channel-number*} | **mac-address** *MAC-address*}	Binds the virtual Fibre Channel interface to the specified interface.
		Note If this is a 10G breakout port, the *slot*/*port* syntax is *QSFP-module*/*port*.
Step 4	(Optional) switch(config-if)# **no bind** {**interface** {**ethernet** *slot*/*port* | **port-channel** *channel-number*} | **mac-address** *MAC-address*}	Unbinds the virtual Fibre Channel interface from the specified interface.
		Note If this is a 10G breakout port, the *slot*/*port* syntax is *QSFP-module*/*port*.
Step 5	(Optional) switch(config)# **no interface vfc** *vfc-id*	Deletes a virtual Fibre Channel interface.

Example

This example shows how to bind a virtual Fibre Channel interface to an Ethernet interface:

```
switch# configure terminal
switch(config)# interface vfc 4
switch(config-if)# bind interface ethernet 1/4
```

This example shows how to bind a virtual Fibre Channel interface to a Cisco Nexus 2232PP Fabric Extender (FEX) Ethernet interface:

```
switch# configure terminal
switch(config)# interface vfc 1001
switch(config-if)# bind interface ethernet 100/1/1
```

This example shows how to bind a virtual Fibre Channel interface to create a vPC:

```
switch# configure terminal
switch(config)# interface vfc 3
```

```
switch(config-if)# bind interface port-channel 1
```

This example shows how to bind a virtual Fibre Channel interface on a Cisco Nexus 2232PP FEX to create a vPC:

```
switch# configure terminal
switch(config)# interface vfc 1001
switch(config-if)# bind interface ethernet 100/1/1
```

Note An error message is displayed if you attempt to bind the interface to a Cisco Nexus FEX that does not support FCoE.

This example shows how to bind a virtual Fibre Channel interface to a MAC address:

```
switch# configure terminal
switch(config)# interface vfc 2
switch(config-if)# bind mac-address 00:0a:00:00:00:36
```

This example shows how to bind a virtual Fibre Channel interface to a Cisco Nexus 2232PP FEX MAC address:

```
switch# configure terminal
switch(config)# interface vfc 1001
switch(config-if)# bind mac-address 00:01:0b:00:00:02
```

This example shows how to delete a virtual Fibre Channel interface:

```
switch# configure terminal
switch(config)# no interface vfc 4
```

Information about 4 Port vPC

In Cisco Nexus 5600 and 6000 Series Switches, the 4-port vPC provides the capability to associate the vFC interface to individual member of a port-channel that has multiple member ports.

Note This feature is supported only for Cisco Nexus 2300 Series Switches that are connected to Cisco Nexus 5600 and 6000 Series Switches.

Guidelines and Limitations

4 port vPC has the following limitations:

- Due to a hardware limitation, this feature is available only on Cisco Nexus 5600 and Cisco Nexus 6000 Series Switches.

- On the Cisco Nexus Series switches, the 4-port VPC feature is available only for 2348UPQ, 2348TQ, 2332TQ Series switches.

- Total number of port-channels and vFC combined on a given Cisco Nexus 2000 Series switch is limited to 64.

- Active-Active FEX and two-layer vPC topologies are not supported.

- Straight-Through FEX and Switch vPC topologies are supported.

  - In Straight-Through FEX, a Cisco Nexus 2300 FEX is connected to Cisco Nexus 5600 or 6000 series switches.

  - In Switch vPC, the host is directly connected to Cisco Nexus 5600 or 6000 series switches.

- 4 port vFC does not support NPV mode, vEth interface, MAC-bound vFC interface, and VE mode.

Configuring vFC Interface

The following steps show how to configure vPC interface to a member port of a multi-member port-channel.

Note You can un-configure a 4-port vPC only after removing member ports from port-channel. You can un-configure only on a single member port-channel.

Procedure

Step 1 Create a multi-member port-channel.

```
switch(config-vlan)#  interface port-channel 500
switch(config-vlan)# [no]fcoe multi-vfc
```

Step 2 Add individual member port to a multi-member port-channel.

```
switch(config-vlan)# interface ethernet 100/1/1
switch(config-vlan)# channel-group 500
switch (config)# interface ethernet 100/1/2
switch(config-if)# channel-group 500
```

Step 3 Associate vPC to a member port of a multi-member port-channel.

```
switch(config)#  interface vfc 10011
switch(config-vlan)# bind interface ethernet 100/1/1
switch(config-vlan)#  interface vfc 10012
switch (config)#  bind interface ethernet 100/1/2
```

Associating a Virtual Fibre Channel Interface to a VSAN

A unique, dedicated VLAN must be configured at every converged access switch to carry traffic for each Virtual Fabric (VSAN) in the SAN (for example, VLAN 1002 for VSAN 1, VLAN 1003 for VSAN 2, and so on). If MST is enabled, a separate MST instance must be used for FCoE VLANs.

Procedure

	Command or Action	**Purpose**
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **vsan database**	Enters VSAN configuration mode.
Step 3	switch(config-vsan)# **vsan** *vsan-id* **interface vfc** *vfc-id*	Configures the association between the VSAN and virtual Fibre Channel interface.
		The VSAN number must map to a VLAN on the physical Ethernet interface that is bound to the virtual Fibre Channel interface.
Step 4	(Optional) switch(config-vsan)# **no vsan** *vsan-id* **interface vfc** *vfc-id*	Disassociates the connection between the VSAN and virtual Fibre Channel interface.

Example

This example shows how to associate a virtual Fibre Channel interface to a VSAN:

```
switch# configure terminal
switch(config)# vsan database
switch(config-vsan)# vsan 2 interface vfc 4
```

Creating an Implicit Virtual Fibre Channel Port Channel Interface

You can create a virtual Fibre Channel (vFC), and implicitly bind it to an Ethernet interface or a port-channel using a single command. For this, the vFC identifier must match the Ethernet interface or port-channel identifier. The Ethernet interface can be a module (slot or port) or a Fabric Extender (FEX) interface (chassis, slot or port).

Note You cannot create an implicit vFC in a breakout port.

Configuring virtual Fibre Channel Interface

Before you begin

- Ensure you have installed the correct license for FCoE.

- Ensure you have enabled FCoE.

Procedure

Step 1 Enter global configuration mode:

switch# **configure terminal**

Step 2 Create a VFC (if it does not already exist):

Additionally, *vfc slot/port* binds the vFC to an Ethernet *slot/port* interface. *vfc chassis/slot/port* binds the vFC to a FEX interface.

switch(config) # **interface vfc** {id | *slot/port* | *chassis/slot/port*}

Step 3 Bring up the vFC interface:

switch(config-if) # **no shutdown**

Step 4 Required: Exit the interface configuration mode:

switch(config-if) # **exit**

Configuring virtual Fibre Channel Interface

This example shows how to implicitly bind a virtual Fibre Channel interface to an Ethernet interface:

```
switch# configure terminal
switch(config)# interface vfc 101/1/1
switch(config-if)# no shutdown
.
.
.
.
Switch# show interface vfc 101/1/1
    vfc101/1/1 is trunking
    Bound interface is Ethernet101/1/1
    Hardware is Ethernet
    Port WWN is 20:00:00:2a:6a:15:d2:7b
    Admin port mode is F, trunk mode is on
    snmp link state traps are enabled
    Port mode is TF
    Port vsan is 600
    Trunk vsans (admin allowed and active) (1,500,600)
    Trunk vsans (up)                       (600)
    Trunk vsans (isolated)                 ()
    Trunk vsans (initializing)             (1,500)
    1 minute input rate 0 bits/sec, 0 bytes/sec, 0 frames/sec
    1 minute output rate 0 bits/sec, 0 bytes/sec, 0 frames/sec
      49 frames input, 5516 bytes
        0 discards, 0 errors
      49 frames output, 5772 bytes
        0 discards, 0 errors
    last clearing of "show interface" counters Fri Oct 30 06:19:33 2015
    Interface last changed at Fri Oct 30 06:19:33 2015
```

Configuring virtual Fibre Channel – Port Channel Interface

Procedure

Step 1   Enter global configuration mode:

switch# **configure terminal**

Step 2   Create a vFC that implicitly binds to the Ethernet port-channel based on its number:

The port number range is from 257 to 4096.

switch(config) # **interface vfc-port-channel** *port number*

Step 3   Bring up the vFC port:

switch(config-if) # **no shutdown**

Step 4   Required: Exit from the current interface configuration mode:

switch(config-if) # **exit**

Configuring virtual Fibre Channel - Port Channel Interface

The example shows how you can create a vFC-port-channel that implicitly binds to Ethernet port-channel:

```
switch# configure terminal
switch(config)# interface vfc-port-channel 300
switch(config-if)# no shutdown
.
.
.
switch# show interface vfc-port-channel 258

    vfc-po258 is trunking
    Bound interface is port-channel258
    Hardware is Ethernet
    Port WWN is 21:01:8c:60:4f:59:31:3f
    Admin port mode is E, trunk mode is on
    snmp link state traps are enabled
    Port mode is TE
    Port vsan is 600
    Trunk vsans (admin allowed and active) (1,100,500,600)
    Trunk vsans (up)                       (600)
    Trunk vsans (isolated)                 ()
    Trunk vsans (initializing)             (1,100,500)
    1 minute input rate 3448 bits/sec, 431 bytes/sec, 4 frames/sec
    1 minute output rate 9064 bits/sec, 1133 bytes/sec, 4 frames/sec
      977735 frames input, 77172556 bytes
        0 discards, 0 errors
      977733 frames output, 205924892 bytes
        0 discards, 0 errors
    last clearing of "show interface" counters Thu Oct 29 06:35:41 2015
    Interface last changed at Thu Oct 29 06:35:41 2015
```

Verifying the Virtual Interface

To display configuration information about virtual interfaces, perform one of the following tasks:

Command	Purpose
switch# **show interface vfc** *vfc-id*	Displays the detailed configuration of the specified Fibre Channel interface.
switch# **show interface brief**	Displays the status of all interfaces.
switch# **show vlan fcoe**	Displays the mapping of FCoE VLANs to VSANs.

This example shows how to display a virtual Fibre Channel interface bound to an Ethernet interface:

```
switch# show interface vfc 3

vfc3 is up

    Bound interface is Ethernet1/37

    Hardware is Virtual Fibre Channel

    Port WWN is 20:02:00:0d:ec:6d:95:3f

    Admin port mode is F, trunk mode is on

    snmp link state traps are enabled

    Port mode is F, FCID is 0x490100

    Port vsan is 931

    1 minute input rate 0 bits/sec, 0 bytes/sec, 0 frames/sec

    1 minute output rate 0 bits/sec, 0 bytes/sec, 0 frames/sec

      0 frames input, 0 bytes

        0 discards, 0 errors

      0 frames output, 0 bytes

        0 discards, 0 errors

    Interface last changed at Thu May 21 04:44:42 2009
```

This example shows how to display a virtual Fibre Channel interface bound to a MAC address:

```
switch# show interface vfc 1001

vfc1001 is down

    Bound MAC is 00:0a:00:00:00:01

    Hardware is Virtual Fibre Channel

    Port WWN is 23:e8:00:0d:ec:6d:95:3f

    Admin port mode is F, trunk mode is on

    snmp link state traps are enabled

    Port vsan is 901

    1 minute input rate 0 bits/sec, 0 bytes/sec, 0 frames/sec

    1 minute output rate 0 bits/sec, 0 bytes/sec, 0 frames/sec

      0 frames input, 0 bytes
```

```
          0 discards, 0 errors
      0 frames output, 0 bytes
          0 discards, 0 errors
```

This example shows how to display the status of all the interfaces on the switch (some output has been removed for brevity):

```
switch# show interface brief
--------------------------------------------------------------------------------
Interface  Vsan   Admin  Admin  Status       SFP    Oper  Oper   Port
                  Mode   Trunk                      Mode  Speed  Channel
                         Mode                             (Gbps)
--------------------------------------------------------------------------------
fc3/1      1      auto   on     trunking     swl    TE    2      --
fc3/2      1      auto   on     sfpAbsent    --     --           --
...
fc3/8      1      auto   on     sfpAbsent    --     --           --
--------------------------------------------------------------------------------
Interface               Status    IP Address      Speed   MTU   Port
                                                                Channel
--------------------------------------------------------------------------------
Ethernet1/1             hwFailure --              --      1500  --
Ethernet1/2             hwFailure --              --      1500  --
Ethernet1/3             up        --              10000   1500  --
...
Ethernet1/39            sfpIsAbsen --             --      1500  --
Ethernet1/40            sfpIsAbsen --             --      1500  --
--------------------------------------------------------------------------------
Interface               Status    IP Address      Speed   MTU
--------------------------------------------------------------------------------
mgmt0                   up        172.16.24.41    100     1500
--------------------------------------------------------------------------------


--------------------------------------------------------------------------------
Interface  Vsan   Admin  Admin  Status       SFP    Oper  Oper   Port
                  Mode   Trunk                      Mode  Speed  Channel
                         Mode                             (Gbps)
--------------------------------------------------------------------------------
vfc 1      1      F      --     down         --     --           --
...
```

This example shows how to display the mapping between the VLANs and VSANs on the switch:

```
switch# show vlan fcoe

VLAN      VSAN      Status
--------  --------  --------
15        15        Operational
20        20        Operational
25        25        Operational
30        30        Non-operational
```

Mapping VSANs to VLANs Example Configuration

The following example shows how to configure the FCoE VLAN and a virtual Fibre Channel interface:

Procedure

Step 1 Enable the associated VLAN and map the VLAN to a VSAN.

```
switch(config)# vlan 200
switch(config-vlan)# fcoe vsan 2
switch(config-vlan)# exit
```

Step 2 Configure the VLAN on a physical Ethernet interface.

```
switch# configure terminal
switch(config)# interface ethernet 1/4
switch(config-if)# spanning-tree port type edge trunk
switch(config-if)# switchport mode trunk
switch(config-if)# switchport trunk allowed vlan 1,200
switch(config-if)# exit
```

Step 3 Create a virtual Fibre Channel interface and bind it to a physical Ethernet interface.

```
switch(config)# interface vfc 4
switch(config-if)# bind interface ethernet 1/4
switch(config-if)# exit
```

Note By default, all virtual Fibre Channel interfaces reside on VSAN 1. If the VLAN to VSAN mapping is to a VSAN other than VSAN 1, then proceed to Step 4.

Step 4 Associate the virtual Fibre Channel interface to the VSAN.

```
switch(config)# vsan database
switch(config-vsan)# vsan 2 interface vfc 4
switch(config-vsan)# exit
```

Step 5 (Optional) Display membership information for the VSAN.

```
switch# show vsan 2 membership
vsan 2 interfaces
         vfc 4
```

Step 6   (Optional) Display the interface information for the virtual Fibre Channel interface.

```
switch# show interface vfc 4

vfc4 is up
Bound interface is Ethernet1/4
Hardware is Virtual Fibre Channel
Port WWN is 20:02:00:0d:ec:6d:95:3f
Port WWN is 20:02:00:0d:ec:6d:95:3f
snmp link state traps are enabled
Port WWN is 20:02:00:0d:ec:6d:95:3f
APort WWN is 20:02:00:0d:ec:6d:95:3f
snmp link state traps are enabled
Port mode is F, FCID is 0x490100
Port vsan is 931
1 minute input rate 0 bits/sec, 0 bytes/sec, 0 frames/sec
1 minute output rate 0 bits/sec, 0 bytes/sec, 0 frames/sec
0 frames input, 0 bytes 0 discards, 0 errors
0 frames output, 0 bytes 0 discards, 0 errors
Interface last changed at Thu Mar 11 04:44:42 2010
```
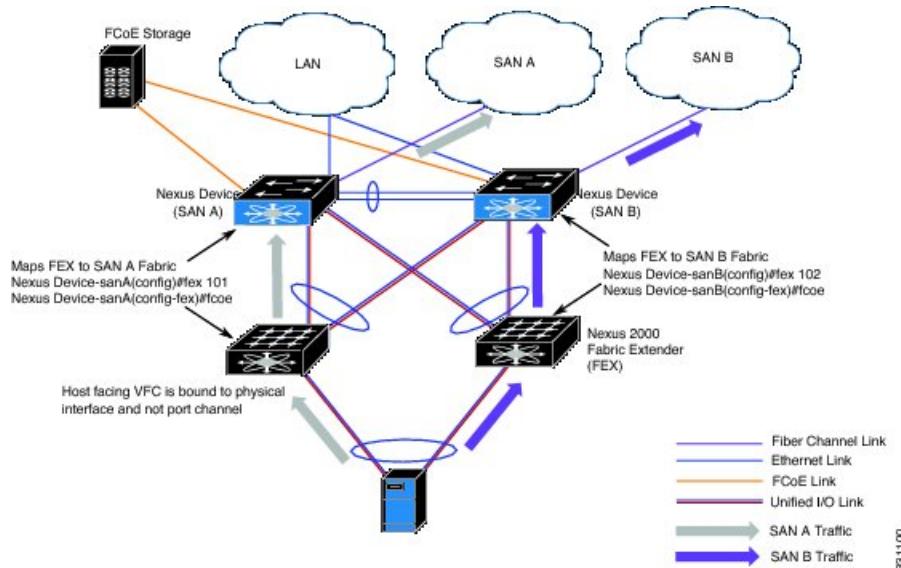
FCoE over Enhanced vPC

Although Ethernet traffic is dual homed between a FEX and a switch pair in an enhanced vPC topology, FCoE traffic must be single-homed to maintain SAN isolation. Therefore, while enhanced vPC supports FCoE, a single homed FEX topology can be a better choice when SAN isolation and high FCoE bandwidth are required.

Consider the following disadvantages of enhanced vPC for a single-homed topology:

- A typical SAN network maintains two fabrics, SAN A and SAN B, with traffic isolated between the two. In an enhanced vPC topology, each switch must be paired (single homed) with a FEX to ensure that FCoE traffic from one FEX is sent to only one switch, while Ethernet traffic is dual homed between each FEX and both switches. Because FCoE traffic from the FEX flows to only one switch while Ethernet traffic flows to both, the traffic load for the FEX uplinks is not evenly balanced.

- In a FEX with eight uplink ports, Ethernet traffic can use all eight ports, while the single-homed FCoE traffic is limited by this topology to using only four of those ports, restricting the maximum bandwidth available for FCoE. As a further restriction, the default QoS template for the shared link allocates only half the link bandwidth to FCoE traffic, with the other half allocated to Ethernet traffic.

- In an enhanced vPC topology with FCoE, the host vPC is limited to two ports, one to each FEX.

The following figure shows the FCoE traffic flow in a system with two Cisco Nexus 2000 FEXs, each associated with a different Cisco Nexus device.

Figure 1: FCoE over Enhanced vPC

Configuring FCoE over Enhanced vPC

FCoE traffic must be single homed to maintain SAN isolation. You must first associate a FEX with only one switch. When the FEX and switch are associated, you can then create a virtual Fibre Channel (vFC) interface and bind it to a port.

After pairing the FEX and switch on the first peer, you repeat the configuration on the second peer using a different port number to ensure SAN traffic isolation. The different configuration will not cause a consistency error because the FCoE portion of the enhanced vPC configuration is not subject to the vPC consistency check.

Before you begin

Review the limitations in FCoE over Enhanced vPC, on page 13.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config) # **fex** *fex-chassis_ID*	Enters configuration mode for the specified FEX.
		The range for *fex-chassis_ID* is from 100 to 199.
Step 3	switch(config-fex) # **fcoe**	Configures the FEX to send FCoE traffic only to this switch.
Step 4	switch(config-fex) # **interface vfc** *vfc-id*	Enters configuration mode for the virtual Fibre Channel interface. If the interface does not

	Command or Action	Purpose
		already exist, this command also creates that interface.
		The range of *vfc-id* is from 1 to 8192.
Step 5	switch(config-if) # **bind interface ethernet** [*fex-chassis-ID/*]*slot/port*	Binds the vFC interface to the specified physical Ethernet interface.
		The range for *fex-chassis_ID* is from 100 to 199. The *slot* must be 1.For FCoE, the range for *port* is from 1 to 32.
		Note If this is a 10G breakout port, the *slot/port* syntax is *QSFP-module/port*.
Step 6	switch(config-if) # **no shutdown**	Returns the interface to its default operational state.
Step 7	(Optional) switch(config-if) # **end**	Return to privileged EXEC mode.
Step 8	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to pair each FEX to a switch for FCoE traffic:

```
nexus5500-sanA# configure terminal
nexus5500-sanA(config) # fex 101
nexus5500-sanA(config-fex) # fcoe
nexus5500-sanA(config-fex) # interface vfc 1
nexus5500-sanA(config-if) # bind interface ethernet 101/1/1
nexus5500-sanA(config-if) # no shutdown
nexus5500-sanA(config-if) # end
nexus5500-sanA# copy running-config startup-config
nexus5500-sanA#

nexus5500-sanB# configure terminal
nexus5500-sanB(config) # fex 102
nexus5500-sanB(config-fex) # fcoe
nexus5500-sanB(config-fex) # interface vfc 1
nexus5500-sanB(config-if) # bind interface ethernet 102/1/1
nexus5500-sanB(config-if) # no shutdown
nexus5500-sanB(config-if) # end
nexus5500-sanB# copy running-config startup-config
nexus5500-sanB#

nexus6000-sanA# configure terminal
nexus6000-sanA(config) # fex 101
nexus6000-sanA(config-fex) # fcoe
nexus6000-sanA(config-fex) # interface vfc 1
nexus6000-sanA(config-if) # bind interface ethernet 101/1/1
nexus6000-sanA(config-if) # no shutdown
```

```
nexus6000-sanA(config-if) # end
nexus6000-sanA# copy running-config startup-config
nexus6000-sanA#

nexus6000-sanB# configure terminal
nexus6000-sanB(config) # fex 102
nexus6000-sanB(config-fex) # fcoe
nexus6000-sanB(config-fex) # interface vfc 1
nexus6000-sanB(config-if) # bind interface ethernet 102/1/1
nexus6000-sanB(config-if) # no shutdown
nexus6000-sanB(config-if) # end
nexus6000-sanB# copy running-config startup-config
nexus6000-sanB#
```

SAN Boot with vPC

A Cisco Nexus Series switch can use SAN boot if the following conditions are met:

- The FEX that contains the port assigned to the vPC must be associated with the Cisco Nexus switch.

- Only one VFC interface is bound to a vPC member. You cannot bind multiple interfaces to multiple members.

Note If you want to ensure backward compatibility for all previous configurations and supported topologies, you must configure the FEX in a straight-through FEX topology that does not use Enhanced vPC.

SAN Boot with vPC Configuration Example

In this example, virtual Fibre Channel interface 1 is bound to physical Ethernet interface 101/1/1 on Fabric A and on interface 102/1/1 on Fabric B. The interface is also associated with virtual port channel 1 on both fabrics.

```
nexus5500-sanA(config) # interface vfc 1
nexus5500-sanA(config-if) # bind interface eth 101/1/1
nexus5500-sanA(config) # interface eth 101/1/1
nexus5500-sanA(config-if) # channel-group 1 mode active
nexus5500-sanA(config-if) # interface port-channel 1
nexus5500-sanA(config-if) # vpc 1
nexus5500-sanA(config-if) #

nexus5500-sanB(config) # interface vfc 1
nexus5500-sanB(config-if) # bind interface eth 102/1/1
nexus5500-sanB(config) # interface eth 102/1/1
nexus5500-sanB(config-if) # channel-group 1 mode active
nexus5500-sanB(config-if) # interface port-channel 1
nexus5500-sanB(config-if) # vpc 1
nexus5500-sanB(config-if) #

nexus5600-sanA(config) # interface vfc 1
nexus5600-sanA(config-if) # bind interface eth 101/1/1
nexus5600-sanA(config) # interface eth 101/1/1
nexus5600-sanA(config-if) # channel-group 1 mode active
nexus5600-sanA(config-if) # interface port-channel 1
```

```
nexus5600-sanA(config-if) # vpc 1
nexus5600-sanA(config-if) #

nexus5600-sanB(config) # interface vfc 1
nexus5600-sanB(config-if) # bind interface eth 102/1/1
nexus5600-sanB(config) # interface eth 102/1/1
nexus5600-sanB(config-if) # channel-group 1 mode active
nexus5600-sanB(config-if) # interface port-channel 1
nexus5600-sanB(config-if) # vpc 1
nexus5600-sanB(config-if) #

nexus6000-sanA(config) # interface vfc 1
nexus6000-sanA(config-if) # bind interface eth 101/1/1
nexus6000-sanA(config) # interface eth 101/1/1
nexus6000-sanA(config-if) # channel-group 1 mode active
nexus6000-sanA(config-if) # interface port-channel 1
nexus6000-sanA(config-if) # vpc 1
nexus6000-sanA(config-if) #

nexus6000-sanB(config) # interface vfc 1
nexus6000-sanB(config-if) # bind interface eth 102/1/1
nexus6000-sanB(config) # interface eth 102/1/1
nexus6000-sanB(config-if) # channel-group 1 mode active
nexus6000-sanB(config-if) # interface port-channel 1
nexus6000-sanB(config-if) # vpc 1
nexus6000-sanB(config-if) #
```