Cisco Nexus 9000 Series NX-OS FC-NPV and FCoE-NPV Configuration Guide, Release 10.2(x)

First Published: 2021-06-30

Last Modified: 2023-09-11

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

CONTENTS

Preface

This preface includes the following sections:

Audience

This publication is for network administrators who install, configure, and maintain Cisco Nexus switches.

Document Conventions

Command descriptions use the following conventions:

Convention	Description
bold	Bold text indicates the commands and keywords that you enter literally as shown.
Italic	Italic text indicates arguments for which you supply the values.
[x]	Square brackets enclose an optional element (keyword or argument).
[x | y]	Square brackets enclosing keywords or arguments that are separated by a vertical bar indicate an optional choice.
{x | y}	Braces enclosing keywords or arguments that are separated by a vertical bar indicate a required choice.
[x {y | z}]	Nested set of square brackets or braces indicate optional or required choices within optional or required elements. Braces and a vertical bar within square brackets indicate a required choice within an optional element.

Convention	Description
`variable`	Indicates a variable for which you supply values, in context where italics cannot be used.
string	A nonquoted set of characters. Do not use quotation marks around the string or the string includes the quotation marks.

Examples use the following conventions:

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

Related Documentation for Cisco Nexus 9000 Series Switches

The entire Cisco Nexus 9000 Series switch documentation set is available at the following URL:

http://www.cisco.com/en/US/products/ps13386/tsd_products_support_series_home.html

Documentation Feedback

To provide technical feedback on this document, or to report an error or omission, please send your comments to nexus9k-docfeedback@cisco.com. We appreciate your feedback.

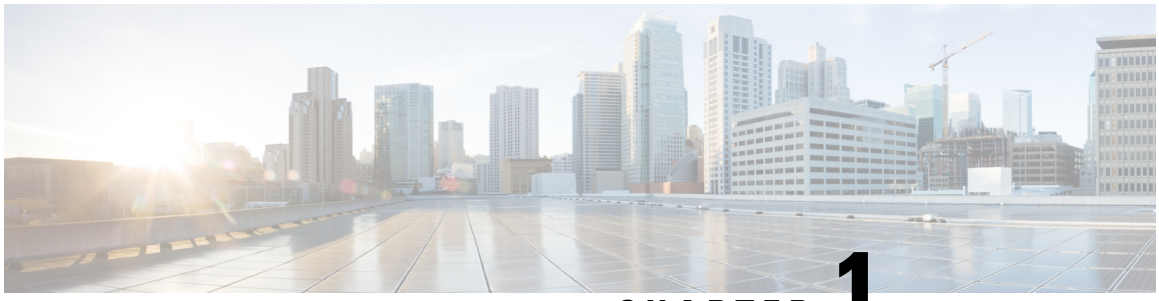
Communications, Services, and Additional Information

- To receive timely, relevant information from Cisco, sign up at Cisco Profile Manager.
- To get the business impact you're looking for with the technologies that matter, visit Cisco Services.
- To submit a service request, visit Cisco Support.
- To discover and browse secure, validated enterprise-class apps, products, solutions and services, visit Cisco Marketplace.
- To obtain general networking, training, and certification titles, visit Cisco Press.
- To find warranty information for a specific product or product family, access Cisco Warranty Finder.

Cisco Bug Search Tool

Cisco Bug Search Tool (BST) is a web-based tool that acts as a gateway to the Cisco bug tracking system that maintains a comprehensive list of defects and vulnerabilities in Cisco products and software. BST provides you with detailed defect information about your products and software.

Preface



CHAPTER 1

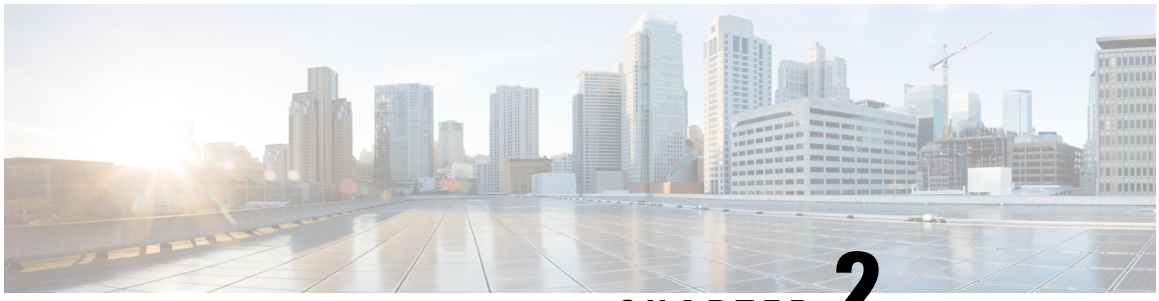
New and Changed Information

New and Changed Information

Table 1: New and Changed Features for Release 10.2(x)

Feature	Description	Changed in Release	Where Documented
FC/FCoE NPV Mode	Added support for Cisco N9K-C9336C-FX2-E platform switches.	10.2(2)F	FC-NPV and FCoE-NPV Supported Hardware, on page 3 Guidelines and Limitations for Configuring FCoE NPV, on page 13 Configuring the FCoE NPV Core Switch, on page 25 Supported Hardware, on page 67 SAN Port Channel Guidelines and Limitations, on page 72 FC NPV Guidelines and Limitations, on page 79 Converting Ethernet Ports to Fibre Channel, on page 82 Configuring FC NP Interfaces, on page 84 FC NPV Core Switch and FC NPV Edge Switch Configuration Example, on page 90

Feature	Description	Changed in Release	Where Documented
There are no new features or enhancements for this release		10.2(1)F	



CHAPTER 2

FC-NPV and FCoE-NPV Supported Hardware

Licensing Requirements

For a complete explanation of Cisco NX-OS licensing recommendations and how to obtain and apply licenses, see the *Cisco NX-OS Licensing Guide* and the *Cisco NX-OS Licensing Options Guide*.

Supported Platforms

Starting with Cisco NX-OS release 7.0(3)I7(1), use the Nexus Switch Platform Support Matrix to know from which Cisco NX-OS releases various Cisco Nexus 9000 and 3000 switches support a selected feature.

FC-NPV and FCoE-NPV Supported Hardware

The following table list the Cisco Nexus 9000 Series hardware that the FC-NPV and FCoE-NPV features are supported.

To enable FC/FCoE NPV mode on Cisco Nexus 9000 series switches, you must configure **feature-set fcoe-npv**.

Note For more information about enabling SAN Switching mode on Cisco Nexus 9000 series switches, see the *Cisco Nexus 9000 Series NX-OS SAN Switching Configuration Guide, Release 10.2(x)*

FC-NPV and FCoE-NPV Supported Hardware

Table 2: Cisco Nexus 9000 FC-NPV and FCoE-NPV Supported Hardware

Switches / Line cards	Model (PID)	FC NP Port	FCoE NP Port	FC Edge Port	FCoE Edge Port	FEX Support
Cisco Nexus 9300 Series Switches	N9K-C93180YC-EX	No	Yes	No	Yes	Yes
	N9K-C93180YC-FX	Yes	Yes	Yes	Yes	Yes
	N9K-C93360YC-FX2	Yes	Yes	Yes	Yes	No
	N9K-C9336C-FX2-E	Yes	Yes	Yes	Yes	No
Cisco Nexus 9504 and 9508 Switches	N9K-X9732C-EX	No	Yes	No	Yes	No
	N9K-X9736C-FX	No	Yes	No	Yes	No

FC-NPV and FCoE-NPV is supported on Cisco N9K-C9336C-FX2-E platform switches.

ISSU with with FCoE (Fiber Channel over Ethernet)/FC (Fiber Channel) NPV (N-port Virtualization) is supported on some Cisco Nexus 9000 switches. An ISSU allows you to upgrade the device software while the switch continues to forward traffic. You can perform an in-service software upgrade (ISSU), also known as a nondisruptive upgrade, for some Cisco Nexus 9000 switches. The default upgrade process is disruptive. Using the nondisruptive option helps ensure a nondisruptive upgrade. (See *Cisco Nexus 9000 Series NX-OS Software Upgrade and Downgrade Guide, Release 10.x* for a complete list of supported platforms).

Note Software Maintenance Upgrade (SMU) is not supported for FC and FCOE.

Table 3: ISSU Support Matrix

Switches / Line cards	Model (PID)	With FC/FCOE NPV	With ST FEX	With AA FEX
Cisco Nexus 9300 Series Switches	N9K-C93180YC-EX	Yes	No	No
	N9K-C93180YC-FX	Yes	No	No
	N9K-C93360YC-FX2	Yes	N/A	N/A
Cisco Nexus 9504 and 9508 Switches	N9K-X9464PX	Yes	Yes	No
	N9K-X9536PQ	Yes	Yes	No
	N9K-X9636PQ	Yes	Yes	No
	N9K-X9732C-EX	No	N/A	N/A
	N9K-X9736C-FX	No	N/A	N/A

FC-NPV is supported on N9k-C93180YC-FX and N9k-C93360YC-FX2 switches and only the following SFPs are supported:

- DS-SFP-FC8G-SW

- DS-SFP-FC16G-SW
- DS-SFP-FC32G-SW

FC-NPV is supported on N9K-C9336C-FX2-E switches and only the following QSFPs are supported:

- DS-SFP-4X32G-SW

FCoE-NPV supports the following FEX:

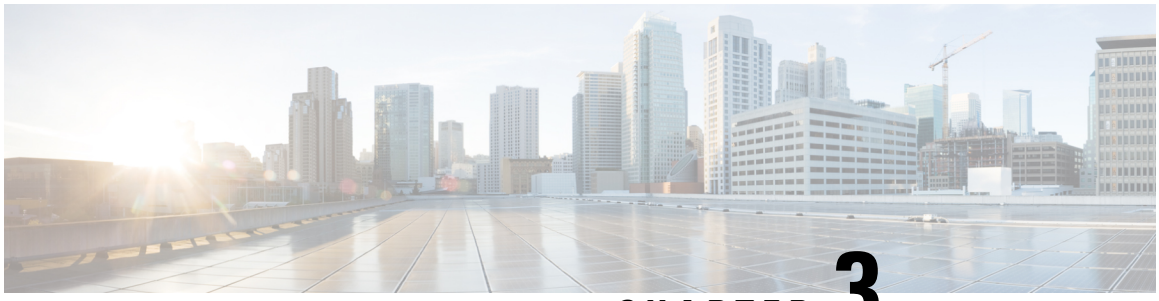
- N2K-B22HP-P
- N2K-B22IBM-P
- N2K-C2232PP
- N2K-C2348UPQ

Note FCoE NPV with FEX is not supported on N9K-C9336C-FX2-E and N9K-C93360YC-FX2.

25G adapter QL41212H is supported on the following devices. However, when a vFC port is shut for more than 60 seconds, the FIP VLAN request retries may not be sent from the host due to a driver issue. For more information about the issue see: CSCvt83152.

- N9K-C93180YC-FX
- N9K-C93180YC-EX
- N9K-C93360YC-FX2
- N9K-C9336C-FX2-E

FC-NPV and FCoE-NPV Supported Hardware



CHAPTER 3

Configuring FCoE NPV

FCoE NPV Overview

Fiber Channel over Ethernet (FCoE) N-port Virtulization (NPV) is an enhanced form of FCoE Initialization Protocol (FIP) snooping that provides a secure method to connect FCoE-capable hosts to an FCoE-capable FCoE forwarder (FCF) device.

FCoE NPV enables:

- The switch to act as an N-port virtualizer (NPV) connected to the core switch (FCF).
- The core switch (FCF) to view the NPV switch as another host.
- The multiple hosts connected to the NPV switch are presented as virtualized N-ports on the core switch (FCF).

FCoE NPV Benefits

FCoE NPV provides the following:

- FCoE NPV provides the advantages of NPV to FCoE deployments (such as preventing domain ID sprawl and reducing Fiber-Channel Forwarder (FCF) table size).
- FCoE NPV provides a secure connect between FCoE hosts and the FCoE FCF.
- FCoE NPV does not have the management and troubleshooting issues that are inherent to managing hosts remotely at the FCF.
- FCoE NPV implements FIP snooping as an extension to the NPV function while retaining the traffic-engineering, VSAN-management, administration, and trouble shooting aspects of NPV.

FCoE NPV Features

The following are the FCoE NPV features:

- Automatic load balance of server logins
  - The server interfaces (Host logins) are distributed in a round robin fashion among the available multiple uplinks (NP ports or external-interfaces).
  - You can enable disruptive automatic load balancing to load balance the existing server interfaces (hosts) to newly added NP uplink interfaces.

    Example:

    ```
    switch(config)# npv auto-load-balance disruptive
    ```

- Traffic mapping
  - You can specify the NP uplinks that a server interface can use to connect to core switches.
  - If the current mapped uplink goes down, the server does not log in through other available uplinks.

    Example:

    ```
    switch(config)# npv traffic-map server-interface vfc2/1 external-interface vfc2/1
    ```

- FCoE forwarding in the FCoE NPV bridge.
- FCoE NPV supports the Data Center Bridging Exchange Protocol (DCBX).
- FCoE frames received over VNP ports are forwarded only if the L2_DA matches one of the FCoE MAC addresses assigned to hosts on the VF ports.

Note FCoE NPV over port channel VNP ports use automatic traffic mapping only for FIP negotiations. FCoE traffic distribution over port channel VNP ports is based on the computed hash value.

Fibre Channel Slow Drain Device Detection and Congestion Avoidance

The data traffic between the end devices in Fibre Channel over Ethernet (FCoE) uses link level and per-hop based flow control. When the slow devices are attached to the fabric, the end devices do not accept the frames at a configured rate. The presence of the slow devices leads to traffic congestion on the links. The traffic

congestion affects the unrelated flows in the fabric that use the same inter-switch links (ISLs) for its traffic, even though the destination devices do not experience the slow drain.

Slow drain device detection and congestion avoidance is supported on below platform switches:

- N9K-C93360YC-FX2
- N9K-C9336C-FX2-E
- N9K-C93180YC-EX
- N9K-X9732C-EX Line Card
- N9K-C93180LC-EX
- N9K-C93180YC-FX
- N9K-X9736C-FX line card

Note Slow drain device detection and congestion avoidance is not supported on FEX ports.

VNP Ports

Connectivity from an FCoE NPV bridge to the FCF is supported only over point-to-point links. These links can be individual Ethernet interfaces or port channel interfaces. For each FCF connected to an Ethernet/port-channel interface, a vFC interface must be created and bound to it. These vFC interfaces must be configured as VNP ports.

On the VNP port, the FCoE NPV bridge emulates an FCoE-capable host with multiple enodes, each with a unique enode MAC address. By default, the VNP port is enabled in trunk mode.

Multiple VSANs can be configured on the VNP port. The FCoE VLANs that correspond to the VNP port VSANs must be configured on the bound Ethernet interface.

Note VNP ports on the Cisco Nexus 9000 Series device emulate an FCoE capable host with multiple Ethernet nodes, each with unique Fabric Provided MAC-Addresses (FPMA).

Licensing Requirements for FCoE NPV

The following table shows the licensing requirements for FCoE NPV:

Product	License Requirement
Cisco NX-OS	FCoE NPV requires the FCoE NPV license (FCOE_NPV_PKG). The PIDs N93-16Y-SSK9 or N93-48Y-SSK9 or ACI-STRG can also be used to enable FCoE NPV along with FC NPV on the supported platforms. For a complete explanation of the Cisco NX-OS licensing scheme and how to obtain and apply the licenses, see the Cisco NX-OS Licensing Guide. **Note** ACI-STRG will license only 48 ports of native Fiber Channel ports. Syslog will not be generated if you use this license on more than 48 ports on Cisco Nexus N9K-C93360YC-FX2 and N9K-C9336C-FX2-E platform switches.

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).

Information About Shutting Down LAN Traffic

Converged Network Adapters (CNA) enable both FCoE and LAN traffic (Unified I/O) to co-exist over a physical link.

In vPC configurations with CNAs, network parameters need to be consistent across peer switches. If the system detects an inconsistency, the secondary vPC leg goes down. Since vPC legs carry both FCOE and LAN traffic, the FCoE link goes down also.

To avoid having the FCoE link go down in this situation, you can use the **shutdown lan** command to shutdown only the LAN traffic on port-channels and individual Ethernet ports.

Note When vPC triggers the vPC secondary leg to be brought down, only the Ethernet VLANs are brought down for the secondary vPC leg. FCoE/storage VLANs of the secondary vPC leg remain up.

Notes About the shutdown lan Command

- The **shutdown lan** command is only configurable on port-channel interfaces, FEX HIF ports, or on individual Ethernet interfaces that vFC interfaces are bound upon.
- The **shutdown lan** command is only configurable on port-channel interfaces or on individual Ethernet interfaces that are in an operational trunking state.
- The **shutdown lan** command cannot be enabled on the secondary vPC leg, if the vPC enabled shutdown lan is applied on the secondary vPC leg.
- A vPC enabled shutdown LAN is not operable if the **shutdown lan** command is applied on the secondary vPC leg.
- The **shutdown lan** command is not configurable on port-channel members.
- The **shutdown lan** command default is **no shutdown lan** (**shutdown lan** is disabled).
- The **shutdown lan** command has a prerequisite that the Link Layer Discovery Protocol (LLDP) feature be enabled.
- A port with a shutdown LAN configuration enabled cannot be added to a port channel.
- The shutdown LAN enable/disable configuration is on a per interface basis.
- If a shutdown lan is configured on an interface, a **no shut** command on the interface does not bring up LAN VLANs.
- A shutdown LAN is triggered when a Type-1 inconsistency occurs in a VPC network.

Examples of Shutdown LAN Traffic

- Shutdown the LAN traffic on port-channel.

```
switch(config)#interface port-channel 955
switch(config-if)# shutdown lan
```

- Shutdown the LAN traffic on individual Ethernet port.

```
switch(config)#interface Ethernet 2/5
switch(config-if)# shutdown lan
```

Examples of Verifying Shutdown LAN Traffic

- Verifying when the **shutdown lan** command is issued on port-channel 955 with Ethernet interface 2/5 as member.

```
switch# sh interface port-channel 955 | grep LAN
All LAN VLANs are administratively shut

switch# sh interface ethernet 2/5 | grep LAN
All LAN VLANs are administratively shut

switch# sh run interface port-channel 955 | grep shut
shutdown lan

switch# sh run interface e2/5 | grep shut
```

```
shutdown lan
```

- Verifying when the vPC triggers shutdown LAN on the secondary vPC leg (port-channel 231 with Ethernet 2/31 as member).

```
switch# sh interface port-channel 231 | grep LAN
All LAN VLANs are administratively shut
```

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,25-Gigabit,40-Gigabit and 100-Gigabit Ethernet interfaces. 10-Gigabit and 25-Gigabit breakout is supported on FCoE 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).
  - You must configure MTU as 9216 or maximum allowed MTU size.

- The vFC interface cannot be bound to Ethernet port channel with multiple member ports connected to FCoE Initialization Protocol (FIP) snooping bridges. It is recommended to use MAC bound vFC when hosts are connected via snooping bridges.
- For VF mode, each vFC interface is associated with only one VSAN.

  For VNP mode, each vFC interface is associated with multiple VSANs.
- 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 different 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.
- FCoE VLANs do not support Multiple Spanning Trees (MST). Creating an MST instance for an FCoE VLAN might cause SAN traffic disruption.

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.

Guidelines and Limitations for Configuring FCoE NPV

Configuring FCoE NPV has the following configuration guidelines and limitations:

- The FCoE NPV on N9K-X9732C-EX and N9K-X9736C-FX line cards is supported only with fabric modules N9K-C9508-FM-E or N9K-C9504-FM-E.
- Enabling FCoE NPV requires:
  - Enabling the LLDP feature using **feature lldp**. LLDP is not enabled by default.
  - Downloading and installing any of the FCOE_NPV licenses.
  - Installing the FCoE-NPV feature set using the **install feature-set fcoe-npv** command.
  - Enabling the FCoE-NPV feature set using the **feature-set fcoe-npv** command. You may have to reload the switch if an existing FCoE feature is enabled.
- Fibre Channel N-port Virtualization (NPV) can co-exist with VXLAN on different fabric uplinks but on same or different front panel ports on the Cisco Nexus 93180YC-FX, N9K-C9336C-FX2-E, and N9k-C93360YC-FX2 switches. If FCOE NPV is installed as a RPM, see the modularity section in the Cisco Nexus 9000 Series NX-OS Software Upgrade and Downgrade Guide for more information.
- Beginning with Cisco NX-OS Release 10.2(2)F, FCoE NPV supports Cisco N9K-C9336C-FX2-E platform switches.
- The first operational port of the port-channel (non-lacp) must be shut down before being removed. Otherwise, the vfc-po binding of that port-channel may go down.
- It is mandatory to do a TCAM Reservation (as mentioned here: Configuring QoS for no-drop Support, on page 18) for FCoE NPV to work.
- The **show** commands with the **internal** keyword are not supported.
- FCoE NPV does not convert server FLOGI to FDISC.
- FCoE NPV supports a VFC port bound to an Ethernet interface, port-channel, or break-out interface.
- FCoE NPV does not support nested NPV.
- FCoE NPV supports FLOGI/FDISC (nested NPIV).
- FCOE is not supported with Copper SFPs.

- To support multiple FLOGI from a single port, you must enable the NPIV feature to accommodate hosts or servers that send FDISC followed by FLOGI.

  Examples of commands to enable/disable and display status of the NPIV feature:

  -
  ```
  switch(config)# feature npiv
  switch# show feature | include npiv
  npiv                  1          enabled
  switch#
  ```

  -
  ```
  switch# show npv status

  npiv is enabled

  disruptive load balancing is disabled

  External Interfaces:
  ====================
    Interface: vfc-po100, State: Trunking
          VSAN:    1, State: Waiting For VSAN Up
          VSAN:    2, State: Up
          VSAN:    3, State: Up, FCID: 0x040000
    Interface: vfc1/49, State: Down

    Number of External Interfaces: 2
  ```

  -
  ```
  switch(config)# no feature npiv
  switch# show feature | include npiv
  npiv                  1          disabled
  switch#
  ```

- When a switch is configured to run non-CoS3 FC/FCoE traffic using User-defined QoS policies, all FC/FCoE interfaces must be configured using the same User-defined QoS input policy.
- FC/FCoE configuration does not support rollback. If FC/FCoE configurations are present, use the best-effort option. All other configurations will be successful, however, error message will be displayed for the FC/FCoE configuration.

Configuring FC/FCoE

Perform TCAM Carving

This section explains how to perform TCAM carving.

```
switch(config)# feature-set fcoe-npv
```

Configure the following (if not configured already) for fcoe-npv to be fully functional:

- hardware access-list tcam region ing-redirect 256
- 256 is the minimum tcam space required in ing-redirect regions for FC/FCoE.

If the required tcam space is not available then ing-racl region can be reduced using the following command:

- hardware access-list tcam region ing-racl 1536

Note 'show hardware access-list tcam region' - Use this command to verify the current tcam configuration.

SUMMARY STEPS

1. Perform TCAM carving.
2. Use the command **show hardware access-list tcam region** to view the configured TCAM region size.
3. Save the configuration and use the command **reload** to reload the switch.

DETAILED STEPS

Step 1 Perform TCAM carving.

Example:

```
Switch(config)# hardware access-list tcam region ing-racl 1536
Switch(config)# hardware access-list tcam region ing-ifacl 256
```

Step 2 Use the command **show hardware access-list tcam region** to view the configured TCAM region size.

Example:

```
Switch(config)# show hardware access-list tcam region
Switch(config)#
```

Step 3 Save the configuration and use the command **reload** to reload the switch.

Example:

```
Switch(config)# reload
Switch(config)#
```

What to do next

You must reload the switch after carving TCAM

Configuring LLDP

This section explains how to configure LLDP.

SUMMARY STEPS

1. **configure terminal**
2. **[no]feature lldp**

DETAILED STEPS

Step 1 **configure terminal**

Enters global configuration mode.

Step 2 [**no**]**feature lldp**

Enables or disables LLDP on the device. LLDP is disabled by default.

Configuring QoS

Configuring Default QoS

There are four types of FCoE default policies: network QoS, output queuing, input queuing, and QoS. You can enable the FCoE default policies by enabling the FCoE NPV feature using the **feature-set fcoe-npv** command. The default QoS ingress policy, **default-fcoe-in-policy**, is implicitly attached to all FC and SAN-port-channel interfaces to enable FC to FCoE traffic; this can be verified by using **show interface** {*fc slot/port* | *san-port-channel <no>*} **all** command. The default QoS policy uses CoS3 and Q1 for all FC and FCoE traffic.

Configuring User Defined QoS

To use a different queue or CoS value for FCoE traffic, create user-defined policies. The user-defined QoS ingress policy has to be created and attached explicitly to both FC and FCoE interfaces to enable traffic to use a different queue or CoS. User-defined QoS policies must be created and activated for system-wide QoS.

The following example demonstrates how to configure and activate user-defined QoS policies that use CoS3 and Q2 for all FC and FCoE traffic.

- Creating a user-defined network QOS policy:

```
switch(config)# policy-map type network-qos fcoe_nq
switch(config-pmap-nqos)# class type network-qos c-nq1
switch(config-pmap-nqos-c)# mtu 1500
switch(config-pmap-nqos-c)# class type network-qos c-nq2
switch(config-pmap-nqos-c)# mtu 9216
switch(config-pmap-nqos-c)# pause pfc-cos 3
switch(config-pmap-nqos-c)# class type network-qos c-nq3
switch(config-pmap-nqos-c)# mtu 1500
switch(config-pmap-nqos-c)# class type network-qos c-nq-default
switch(config-pmap-nqos-c)# mtu 1500
switch(config-pmap-nqos-c)# exit
switch(config-pmap-nqos)# exit
switch(config)#
```

- Creating a user-defined input queuing policy:

```
switch(config)# policy-map type queuing fcoe-in-policy
switch(config-pmap-que)# class type queuing c-in-q2
switch(config-pmap-c-que)# bandwidth percent 50
```

```
switch(config-pmap-c-que)# class type queuing c-in-q-default
switch(config-pmap-c-que)# bandwidth percent 50
switch(config-pmap-c-que)# exit
switch(config-pmap-que)# exit
switch(config)
```

- Creating a user-defined output queuing policy:

```
switch(config)# policy-map type queuing fcoe-out-policy
switch(config-pmap-que)# class type queuing c-out-q3
switch(config-pmap-c-que)# priority level 1
switch(config-pmap-c-que)# class type queuing c-out-q-default
switch(config-pmap-c-que)# bandwidth remaining percent 50
switch(config-pmap-c-que)# class type queuing c-out-q1
switch(config-pmap-c-que)# bandwidth remaining percent 0
switch(config-pmap-c-que)# class type queuing c-out-q2
switch(config-pmap-c-que)# bandwidth remaining percent 50
switch(config-pmap-c-que)# exit
switch(config-pmap-que)# exit
switch(config)#
```

- Creating a user-defined QoS input policy:

```
switch(config)# class-map type qos match-any fcoe
switch(config-cmap-qos)# match protocol fcoe
switch(config-cmap-qos)# match cos 3
switch(config-cmap-qos)# exit
switch(config)#
switch(config)# policy-map type qos fcoe_qos_policy
switch(config-pmap-qos)# class fcoe
switch(config-pmap-c-qos)# set cos 3
switch(config-pmap-c-qos)# set qos-group 2
switch(config-pmap-c-qos)# exit
switch(config-pmap-qos)# exit
switch(config)#
```

- Activating a user-defined system QoS policy:

```
switch(config)# system qos
switch(config-sys-qos)# service-policy type queuing input fcoe-in-policy
switch(config-sys-qos)# service-policy type queuing output fcoe-out-policy
switch(config-sys-qos)# service-policy type network-qos fcoe_nq
switch(config-sys-qos)# exit
switch(config)#
```

- Applying a QoS input policy to an FC or FCoE interface:

```
switch# conf
switch(config)# interface fc <slot>/<port> | ethernet <slot>/<port> | san-port-channel
 <no> | port-channel <no>
switch(config-if)# service-policy type qos input fcoe_qos_policy
```

- Removing a QoS input policy from an FC or FCoE interface:

```
switch# conf
switch(config)# interface fc <slot>/<port> | ethernet <slot>/<port> | san-port-channel
 <no> | port-channel <no>
switch(config-if)# no service-policy type qos input fcoe_qos_policy
```

- Verifying a QoS input policy applied to an FC or FCoE interface:

```
switch# show running-config interface fc <slot>/<port> | interface <slot>/<port> |
san-port-channel <no> | port-channel <no> all
```

Note

- When a user-defined QoS policy is used, the same QoS input policy must be applied to all FC and FCoE interfaces in the switch.
- Do not configure **match protocol fcoe** under more than one QoS class map, as FCoE traffic is supported only on a single CoS.

Configuring Traffic Shaping

Traffic shaping is used to control access to available bandwidth and to regulate the flow of traffic in order to avoid congestion that can occur when the sent traffic exceeds the access speed. Because traffic shaping limits the rate of transmission of data, you may use this command only when necessary.

The following example demonstrates how to configure traffic shaper:

- The following command displays the default system level settings for all FC interfaces:

```
switch(config)# show running-config all | i i rate
hardware qos fc rate-shaper
switch(config)#
```

- The following example shows how to configure rate shaper. This command is applied on all FC interfaces:

Note Rarely, you may see input discards on any of the 4G, 8G, 16G, or 32G interfaces. Use the command *hardware qos fc rate-shaper [low]*, to configure the rate shape. Because this is a system level configuration, it will apply to all the FC ports and will reduce the rates for all FC ports. The default option of the command *hardware qos fc rate-shaper* is applicable to all FC interfaces.

```
switch(config)# hardware qos fc rate-shaper low
switch(config)#
switch(config)#end
```

Configuring QoS for no-drop Support

A qos ingress policy is used to mark ingress FCoE frames. The qos ingress policy must be applied to the interfaces that handle FCoE traffic (such as, all ethernet/port-channel interfaces bound to vFCs).

Note Check to ensure that the port qos region has hardware TCAM space reserved.

This step is mandatory for FCoE NPV to work.

- Reserve TCAM space for the QoS region.

  You may need to acquire TCAM space reserved for other regions (such as the l3qos region).

- Save the configuration.
- Reload the line cards or switch.
- Confirm the port qos region TCAM space.
- Example for TCAM carving on N9K-C93180YC-EX, N9K-C93180YC-FX, N9K-C93360YC-FX2, or N9K-C9336C-FX2-E:

```
hardware access-list tcam region ing-racl 1536
hardware access-list tcam region ing-redirect 256
```

Example:

```
switch# show hardware access-list tcam region | i "IPV4 Port QoS \[qos\] size"
IPV4 Port QoS [qos] size = 0 /*** Value is 0; No reserved TCAM space.***/

switch# config
switch(config)# hardware access-list tcam region qos 256

Warning: Please reload all linecards for the configuration to take effect

switch# copy running-config startup-config

switch# reload

switch# show hardware access-list tcam region | i "IPV4 Port QoS \[qos\] size"
IPV4 Port QoS [qos] size = 256
```

Configuring FCoE QoS policies

- There are four types of FCoE default policies: network-qos, output queuing, input queuing, and qos.
- You can activate the FCoE default policies by enabling the FCoE-NPV feature using the **feature-set fcoe-npv** command and remove the FCoE default policies by executing the **no feature-set fcoe-npv** command.
- Before entering **no feature-set fcoe-npv**, remove all FCoE policies from the interface and system level. The **no feature-set fcoe-npv** command is allowed only when there are no FC ports configured.

Note Cisco recommends using the FCoE default policies. All policies applied must be of the same type, either 4q or 8q mode, and must be explicitly applied or removed at the system and interface level.

- When configuring QoS policies for an active-active FEX topology that is enabled for FCoE, you must configure the QoS policies on the FEX HIF port on both VPC peers to avoid unpredictable results.
- To use a different queue or cos value for FCoE traffic, create user-defined policies.

Configuring QoS Policies for FCoE

- You can configure a QoS policy by following one of these methods:
  - Predefined policies—You can apply a predefined QoS policy: **default-fcoe-in-policy**.

Note No policy will be applied by default for FCoE.

  - User-defined policy—You can create a QoS policy that conforms to one of the system-defined policies.

Configuring System-wide QoS Policy

Note The network-qos policy and output/input queuing policies should be applied at the system level and the qos policy should be applied at the interface level, for every interface that carries the FCoE traffic.

```
switch(config)# system qos
switch(config-sys-qos)# service-policy type queuing input default-fcoe-in-que-policy
switch(config-sys-qos)# service-policy type queuing output { default-fcoe-8q-out-policy |
default-fcoe-out-policy }
switch(config-sys-qos)# service-policy type network-qos { default-fcoe-8q-nq-policy |
default-fcoe-nq-policy }
```

Configuration Example for user-defined policies

```
switch(config)# policy-map type network-qos fcoe_nq
switch(config-pmap-nqos)# class type network-qos c-nq1
switch(config-pmap-nqos-c)# pause pfc-cos 3
switch(config-pmap-nqos-c)# mtu 9216
switch(config-pmap-nqos-c)# class type network-qos c-nq2
switch(config-pmap-nqos-c)# mtu 1500
switch(config-pmap-nqos-c)# class type network-qos c-nq3
switch(config-pmap-nqos-c)# mtu 1500
switch(config-pmap-nqos-c)# class type network-qos c-nq-default
switch(config-pmap-nqos-c)# mtu 1500
switch(config-pmap-nqos-c)# exit
switch(config-pmap-nqos)# exit
switch(config)#
switch(config)# policy-map type queuing fcoe-in-policy
switch(config-pmap-que)# class type queuing c-in-q1
switch(config-pmap-c-que)# bandwidth percent 50
switch(config-pmap-c-que)# class type queuing c-in-q-default
switch(config-pmap-c-que)# bandwidth percent 50
switch(config-pmap-c-que)# exit
switch(config)
switch(config)# policy-map type queuing fcoe-out-policy
switch(config-pmap-que)# class type queuing c-out-q3
```

```
switch(config-pmap-c-que)# priority level 1
switch(config-pmap-c-que)# class type queuing c-out-q-default
switch(config-pmap-c-que)# bandwidth remaining percent 50
switch(config-pmap-c-que)# class type queuing c-out-q1
switch(config-pmap-c-que)# bandwidth remaining percent 50
switch(config-pmap-c-que)# class type queuing c-out-q2
switch(config-pmap-c-que)# bandwidth remaining percent 0
switch(config-pmap-c-que)# exit
switch(config)#
switch(config)# class-map type qos match-any fcoe
switch(config-cmap-qos)# match protocol fcoe
switch(config-cmap-qos)# match cos 3
switch(config-cmap-qos)# exit
switch(config)#
switch(config)# policy-map type qos fcoe_qos_policy
switch(config-pmap-qos)# class fcoe
switch(config-pmap-c-qos)# set cos 3
switch(config-pmap-c-qos)# set qos-group 1
switch(config-pmap-c-qos)# exit
switch(config-pmap-qos)# exit
switch(config)#
switch(config)# system qos
switch(config-sys-qos)# service-policy type queuing input fcoe-in-policy
switch(config-sys-qos)# service-policy type queuing output fcoe-out-policy
switch(config-sys-qos)# service-policy type network-qos fcoe_nq
```

Note The **set cos 3** command under the QOS policy is mandatory only when there are native fiber channel ports and the command is applicable only for N9K-C93180YC-FX, N9K-C9336C-FX2-E , and N9k-C93360YC-FX2 platforms. For all the other Cisco Nexus 9000 Platform switches, this step is optional.

Note When FEX is connected:

- Apply the QoS policy to the system level and to the HIF port to honor the pause frames in the FCoE traffic.
- 8q policies are not supported when FEX is online.

```
switch(config)# system qos
switch(config-sys-qos)# service-policy type queuing input policy-name
switch(config-sys-qos)# service-policy type queuing output policy-name
switch(config-sys-qos)# service-policy type network-qos policy-name
switch(config-sys-qos)# service-policy type qos input policy-name
```

Applying the ingress QoS policy to each Ethernet/port-channel interface that is bound to vFC interface for FCoE.

```
switch(config)# interface ethernet 2/1
switch(config-if)# switchport mode trunk
switch(config-if)# mtu 9216 /* Or maximum allowed value */
switch(config-if)# service-policy type qos input { default-fcoe-in-policy | fcoe_qos_policy
 }
switch(config-if)# exit
switch(config)#
```

Note The QoS policy needs to be attached to an HIF interface or the port-channel of an HIF interface:

- HIF interface

```
interface "HIF port"
service-policy type qos input policy-name
```

- Port-channel of an HIF interface

```
interface port-channel
service-policy type qos input policy-name
```

Configuring FCoE NPV

Configuring VLAN-VSAN Mapping

VSANs and VLANs are required and the VSANs need to be mapped to the VLANs.

One VLAN can be mapped to only one VSAN and vice versa. The VSANs can then be added to F and NP vFC interfaces (described in a subsequent section).

- Example of VSAN creation:

```
switch(config)#
switch(config)# vsan database
switch(config-vsan-db)# vsan 10
switch(config-vsan-db)#
```

- Example VLAN configuration and binding to FCoE VSAN:

```
switch(config)# vlan 10
switch(config-vlan)# fcoe vsan 10
switch(config-vlan)# exit
switch(config)#
```

Binding vFC to MAC Address

A MAC address bound vFC can also be created on the device interface.

Note A MAC bound vFC can be configured to a host sitting behind a FIP Snooping Bridge (FSB).

When both MAC bound vFC and port-bound vFC are configured for the same interface, the port-bound vFC takes precedence.

As a best practice, you should have either a MAC bound vFC or a port- bound vFC for a physical Ethernet port or a port-channel. However, you cannot have both.

SUMMARY STEPS

1. **configure terminal**
2. **interface vfc** <*number*>
3. **bind mac-address** <*mac-address*>

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**interface vfc** <*number*>	Creates a virtual fibre Channel Interface.
Step 3	**bind mac-address** <*mac-address*>	Binds the MAC address.

Example

The following 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
```

Explicit vFC Configuration

An explicit vFC interface is a vFC interface where the bound ethernet/port-channel interface is explicitly configured. (The interface ID range is 1-8192.)

Note The port VSAN of the vFC and the native VLAN of the ethernet port should not be mapped to each other in a VLAN-VSAN mapping; this will break the FCoE path completely.

- Example of explicit vFC bound to interface Ethernet:

```
switch# configure terminal
switch(config)# interface vfc 21
switch(config-if)# bind interface ethernet 2/1
```

- Example of explicit vFC bound to interface port-channel:

```
switch# configure terminal
switch(config)# interface vfc 100
switch(config-if)# bind interface port-channel 100
```

- Example of explicit vFC bound to break-out port:

```
switch# configure terminal
```

```
switch(config)# interface vfc 111
switch(config-if)# bind interface ethernet 1/1/1
```

- Example of NP interface configuration using explicit vFC:

```
switch# configure terminal
switch(config)# interface vfc21
switch(config-if)# switchport mode NP
switch(config-if)# switchport trunk allowed vsan 10 /* optional; for restricting VSANs
 */
```

- Example of NP interface configuration using explicit bound port-channel interface:

```
switch# configure terminal
switch(config)# interface vfc152
switch(config-if)# bind interface port-channel152
switch(config-if)# switchport mode NP
switch(config-if)# switchport trunk allowed vsan 2
switch(config-if)# switchport trunk mode on
switch(config-if)# no shutdown
```

- Example of F interface configuration using explicit vFC:

```
switch# configure terminal
switch(config)# interface vfc15
switch(config-if)# bind interface ethernet 1/5
switch(config-if)# switchport mode F /* Default mode is F */
switch(config-if)# switchport trunk allowed vsan 10
switch (config-if)# exit
switch (config)# vsan database
switch(config-vsan-db)# vsan 10 interface vfc15
switch(config-vsan-db)# exit
```

Implicit vFC Configuration

An implicit vFC interface is a vFC interface that has an ID with the format *slot*/*port* or *unit*/*slot*/*port* or **port-channel** *id*. When this vFC is created, the Ethernet interface *slot*/*port* or *unit*/*slot*/*port* or **port-channel** *id* is automatically (implicitly) bound to the interface. The running configuration displays the bound Ethernet/port-channel interface. If the Ethernet /port-channel interface does not exist or it is bound to another explicit vFC interface, the vFC creation fails with an error.

Note

- The port VSAN of the vFC and the native VLAN of the Ethernet port should not be mapped to each other in a VLAN-VSAN mapping. It breaks the FCoE path completely.
- When a vFC is created through the Cisco DCNM (Data Center Network Manager), the vFC interface goes to VSAN 4094 (isolated), whereas when a vFC is created through the CLI, the vFC interface goes to VSAN 1. The ethernet interface should be up before configuring implicit vFC through the Cisco DCNM because once the vFC goes to VSAN 4094, it cannot be brought up.

- Example of implicit vFC bound to interface Ethernet:

```
switch# configure terminal
```

```
switch(config)# interface vfc 2/1
```

- Example of implicit vFC bound to interface port-channel:

```
switch# configure terminal
switch(config)# interface vfc-port-channel 100
```

- Example of implicit vFC bound to break-out port:

```
switch# configure terminal
switch(config)# interface vfc 1/1/1
```

- Example of NP interface configuration using implicit vFC:

```
switch# configure terminal
switch(config)# interface vfc1/1/1
switch(config-if)# switchport mode NP
switch(config-if)# switchport trunk allowed vsan 10 /* optional; for restricting VSANs
 */
```

- Example of F interface configuration using implicit vFC:

```
switch# configure terminal
switch(config)# interface vfc1/1/1
switch(config-if)# switchport mode F /* Default mode is F */
switch(config-if)# switchport trunk allowed vsan 10
switch (config-if)# exit
switch (config)# vsan database
switch(config-vsan-db)# vsan 10 interface vfc1/1/1
switch(config-vsan-db)# exit
```

Configuring the FCoE NPV Core Switch

Perform the following steps to configure an FCoE NPV core switch.

SUMMARY STEPS

1. **configure terminal**
2. (Optional) **switchto vdc** *vdc-name*
3. **feature npiv**
4. (Optional) **feature fport-channel-trunk**
5. **interface ethernet** *slot/port*
6. **switchport**
7. **no switchport**
8. **switchport mode trunk**
9. **mtu 9216**
10. **service-policy type** {**network-qos** | **qos** | **queuing**} [**input** | **output**] *fcoe default policy-name*
11. **exit**
12. **interface vfc** *vfc-id*
13. **switchport mode f**

14. **bind interface ethernet** *slot/port*
15. **exit**
16. **vsan database**
17. **vsan** *vsan-id*
18. **vsan** *vsan-id* **interface vfc** *vfc-id*
19. **exit**
20. **vlan** *vlan-id*
21. **fcoe vsan** *vsan-id*
22. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal**	Enters configuration mode.
Step 2	(Optional) **switchto vdc** *vdc-name*	Switch to storage VDC. **Note** This step is required only when a Cisco Nexus 7000 Series switch is used as the core switch.
Step 3	**feature npiv**	Enable NPIV.
Step 4	(Optional) **feature fport-channel-trunk**	Enables F port channel trunking.
Step 5	**interface ethernet** *slot/port*	Enters interface configuration mode.
Step 6	**switchport**	Configures the interface as a Layer 2 interface and deletes any configuration specific to Layer 3 on this interface.
Step 7	**no switchport**	Configures the interface as a Layer 3 interface and deletes any configuration specific to Layer 2 on this interface.
Step 8	**switchport mode trunk**	Set physical interface mode to trunk.
Step 9	**mtu 9216**	Configure MTU as 9216. You must configure MTU as 9216 or maximum allowed MTU size **Note** This step is required only when a Cisco Nexus N9K-C93180YC-FX, N9K-C9336C-FX2-E or N9K-C93360YC-FX2 switch is used as the core switch.
Step 10	**service-policy type** {**network-qos** | **qos** | **queuing**} [**input** | **output**] *fcoe default policy-name*	Specifies the QoS policy on the port to a no drop policy. **Note** This step is required only when a Cisco Nexus N9K-C93180YC-FX, N9K-C9336C-FX2-E or N9K-C93360YC-FX2 switch is used as the core switch.

	Command or Action	Purpose
Step 11	**exit**	Exits the interface mode.
Step 12	**interface vfc** *vfc-id*	Enters interface configuration mode.
Step 13	**switchport mode f**	Set vFC port mode to VF.
Step 14	**bind interface ethernet** *slot/port*	Binds a ethernet interface to a vFC. **Important** The **bind interface ethernet** command is not required for an implicit vFC configuration.
Step 15	**exit**	Exits the interface configuration mode.
Step 16	**vsan database**	Enters VSAN configuration mode.
Step 17	**vsan** *vsan-id*	Create VSAN
Step 18	**vsan** *vsan-id* **interface vfc** *vfc-id*	Add vFC into VSAN.
Step 19	**exit**	Exits the VSAN configuration mode.
Step 20	**vlan** *vlan-id*	Enters VLAN configuration mode.
Step 21	**fcoe vsan** *vsan-id*	Creates FCoE VLAN and map FCoE VLAN to VSAN.
Step 22	**exit**	Exits the VLAN configuration mode.

Configuring the FCoE NPV Edge Switch

Perform the following steps to configure an FCoE NPV edge switch.

SUMMARY STEPS

1. **install feature-set fcoe-npv**
2. **feature-set fcoe-npv**
3. **[no] feature lldp**
4. **vsan database**
5. **vsan** *vsan-id*
6. **exit**
7. **vlan** *vlan-id*
8. **fcoe vsan** *vsan-id*
9. **exit**
10. **interface ethernet** *slot/port*
11. **switchport**
12. **switchport mode trunk**
13. **mtu 9216**
14. **service-policy type** {**network-qos** | **qos** | **queuing**} [**input** | **output**] *fcoe default policy-name*
15. **exit**

16. **interface vfc** *vfc-id*
17. **switchport mode NP**
18. **bind interface ethernet** *slot/port*
19. **exit**
20. **interface ethernet** *slot/port*
21. **switchport**
22. **switchport mode trunk**
23. **mtu 9216**
24. **service-policy type** {**network-qos** | **qos** | **queuing**} [**input** | **output**] *fcoe default policy-name*
25. **exit**
26. **interface vfc** *vfc-id*
27. **switchport mode f**
28. **switchport trunk mode on**
29. **switchport trunk allowed vsan** *vsan-id*
30. **bind interface ethernet** *slot/port*
31. **no shutdown**
32. **exit**
33. **vsan database**
34. **vsan** *vsan-id* **interface vfc** *vfc-id*
35. **vsan** *vsan-id* **interface vfc** *vfc-id*
36. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**install feature-set fcoe-npv**	Installs FCoE NPV.

Step 2	**feature-set fcoe-npv**	Enables FCoE NPV.
		Note When enabling FCoE NPV for Cisco NX-OS 7.0(3)I4(1) and later releases, the following BCM settings are required per FCoE VLAN:
		`LEARN_DISABLE=1`
		`L2_NON_UCAST_DROP=1`
		`L2_MISS_DROP=1`
		• Ethernet VLANs do not require these BCM settings.

Step 3	**[no] feature lldp**	Enables or disables LLDP on the device. LLDP is disabled by default.
Step 4	**vsan database**	Enters VSAN configuration mode.
Step 5	**vsan** *vsan-id*	Creates VSAN.
Step 6	**exit**	Exits the VSAN configuration mode.
Step 7	**vlan** *vlan-id*	Enters VLAN configuration mode and creates the VLAN.

	Command or Action	Purpose
Step 8	**fcoe vsan** *vsan-id*	Maps the FCoE VLAN to VSAN.
Step 9	**exit**	Exits the VLAN configuration mode.
Step 10	**interface ethernet** *slot/port*	Enters interface configuration mode.
Step 11	**switchport**	To put an interface that is in Layer 3 mode into Layer 2 mode for Layer 2 configuration, use the **switchport** command in interface configuration mode. To put an interface into Layer 3 mode, use the **no** form of this command.
Step 12	**switchport mode trunk**	Sets switch side physical interface to trunk mode.
Step 13	**mtu 9216**	Configure MTU as 9216. You must configure MTU as 9216 or maximum allowed MTU size
Step 14	**service-policy type** {**network-qos** | **qos** | **queuing**} [**input** | **output**] *fcoe default policy-name*	Specifies the QoS policy on the port to a no drop policy.
Step 15	**exit**	Exits the interface configuration mode.
Step 16	**interface vfc** *vfc-id*	Enters interface configuration mode.
Step 17	**switchport mode NP**	Sets vFC port mode to VNP.
Step 18	**bind interface ethernet** *slot/port*	Binds the ethernet interface to vFC. **Important** The **bind interface ethernet** command is not required for an implicit vFC configuration.
Step 19	**exit**	Exits the interface configuration mode.
Step 20	**interface ethernet** *slot/port*	Enters interface configuration mode.
Step 21	**switchport**	To put an interface that is in Layer 3 mode into Layer 2 mode for Layer 2 configuration, use the **switchport** command in interface configuration mode. To put an interface into Layer 3 mode, use the **no** form of this command.
Step 22	**switchport mode trunk**	Sets server side physical interface to trunk mode.
Step 23	**mtu 9216**	Configure MTU as 9216.
Step 24	**service-policy type** {**network-qos** | **qos** | **queuing**} [**input** | **output**] *fcoe default policy-name*	Specifies the default FCoE policy map to use as the service policy for the system.
Step 25	**exit**	Exits the interface configuration mode.
Step 26	**interface vfc** *vfc-id*	Enters interface configuration mode.
Step 27	**switchport mode f**	Sets the mode to F on a Fibre Channel interface.

	Command or Action	Purpose
Step 28	**switchport trunk mode on**	Sets server side physical interface to trunk mode.
Step 29	**switchport trunk allowed vsan** *vsan-id*	Configure vFC port to allow VSAN vsan-id.
Step 30	**bind interface ethernet** *slot/port*	Binds the ethernet interface to vFC. **Important** The **bind interface ethernet** command is not required for an implicit vFC configuration.
Step 31	**no shutdown**	Keeps the Fibre Channel interface active
Step 32	**exit**	Exits the interface configuration mode.
Step 33	**vsan database**	Enters VSAN configuration mode.
Step 34	**vsan** *vsan-id* **interface vfc** *vfc-id*	Adds port VSAN vsan-id to VF port.
Step 35	**vsan** *vsan-id* **interface vfc** *vfc-id*	Add VNP port to VSAN vsan-id. **Note** This step is optional. The default port VSAN is 1 and is preferable for the VNP port.
Step 36	**exit**	Exits the VSAN configuration mode.

Configuring a Pause Frame Timeout Value

You can enable or disable a pause frame timeout value on a port. The system periodically checks the ports for a pause condition and enables a pause frame timeout on a port if it is in a continuous pause condition for a period of time. This situation results in all frames that come to that port getting dropped in the egress. This function empties the buffer space in the ISL link and helps to reduce the fabric slowdown and the congestion on the other unrelated flows using the same link.

Note Configuring a pause frame timeout value is supported on the following switches and line cards:

- N9K-C93360YC-FX2
- N9K-C93180YC-EX
- N9K-C93180YC-FX
- N9K-C93180LC-EX
- N9K-X9732C-EX Line Card
- N9K-X9736C-FX Line Card
- N9K-C9336C-FX2-E

When a pause condition is cleared on a port or when a port flaps, the system disables the pause frame timeout on that particular port.

The pause frame timeout is disabled by default. We recommend that you retain the default configuration for the ISLs and configure a value that does not exceed the default value for the edge ports.

For a faster recovery from the slow drain device behavior, you should configure a pause frame timeout value because it drops all the frames in the edge port that face the slow drain whether the frame is in the switch for a congested timeout or not. This process instantly clears the congestion in the ISL.

Use the **no system default interface pause mode edge** command to disable the pause frame timeout value on the edge ports. The default pause timeout value is 500 milliseconds.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch# **system default interface pause timeout** *milliseconds* **mode edge**
3. switch# **system default interface pause mode edge**
4. switch# **no system default interface pause timeout** *milliseconds* **mode edge**
5. switch# **no system default interface pause mode edge**

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch# **system default interface pause timeout** *milliseconds* **mode edge**	Configures a new pause frame timeout value in milliseconds and the port mode for the device. **Note** Timeout value is specified in multiples of 100 (range is 100-500). **Note** The **system default interface pause timeout** *milliseconds* **mode core** command is not supported.
Step 3	switch# **system default interface pause mode edge**	Configures the default pause frame timeout value in milliseconds and the port mode for the device. **Note** Only the **system default interface pause** *milliseconds* **mode edge** command is supported. The **system default interface pause** *milliseconds* **mode core** command is not supported.
Step 4	switch# **no system default interface pause timeout** *milliseconds* **mode edge**	Disables the pause frame timeout for the device.
Step 5	switch# **no system default interface pause mode edge**	Disables the default pause frame timeout for the device.

Example

This example shows how to configure a pause frame timeout value:

```
switch# configure terminal
switch(config)# system default interface pause timeout 500 mode edge
switch(config)# system default interface pause mode edge
switch(config)# no system default interface pause timeout 500 mode edge
switch(config)# no system default interface pause mode edge
switch(config)# end
```

This example shows how to display pause frame timeout information:

```
switch#(config-if)# attach module 1
module-1# sh creditmon interface ethernet 1/35

Ethernet1/35: PORT is EDGE, xoff_hits=2
         flush-status        : OFF
         total_xoff_hits     : 2
         (cntr) pause frames : 832502
         (cntr) pause quanta : 1962909 milli-seconds
         (cntr) force drops  : 94320764
         (cntr-pg) to_drops  : 0
         DBG_xoff_hit_cnt    : 0
         DBG_xoff_hit_time   : 274
         DBG_port_fc_mode    : 2
         DBG_force_tmo_val   : 300 milli-seconds
         CFG_congestion_tmo  : 0 milli-seconds
```

This example shows how to display pause frame timeout information:

```
switch(config-if)# attach module 1
module-1#
module-1# sh creditmon interface all
Ethernet1/1: PORT is NONE, xoff_hits=0
Ethernet1/2: PORT is NONE, xoff_hits=0
Ethernet1/3: PORT is NONE, xoff_hits=0
Ethernet1/4: PORT is NONE, xoff_hits=0
Ethernet1/5: PORT is NONE, xoff_hits=0
Ethernet1/6: PORT is NONE, xoff_hits=0
Ethernet1/7: PORT is NONE, xoff_hits=0
Ethernet1/8: PORT is NONE, xoff_hits=0
Ethernet1/9: PORT is NONE, xoff_hits=0
Ethernet1/10: PORT is NONE, xoff_hits=0
Ethernet1/11: PORT is NONE, xoff_hits=0
Ethernet1/12: PORT is NONE, xoff_hits=0
Ethernet1/13: PORT is NONE, xoff_hits=0
Ethernet1/14: PORT is NONE, xoff_hits=0
Ethernet1/15: PORT is NONE, xoff_hits=0
Ethernet1/16: PORT is NONE, xoff_hits=0
Ethernet1/17: PORT is NONE, xoff_hits=0
Ethernet1/18: PORT is NONE, xoff_hits=0
Ethernet1/19: PORT is NONE, xoff_hits=0
Ethernet1/20: PORT is NONE, xoff_hits=0
Ethernet1/21: PORT is NONE, xoff_hits=0
Ethernet1/22: PORT is NONE, xoff_hits=0
Ethernet1/23: PORT is NONE, xoff_hits=0
Ethernet1/24: PORT is NONE, xoff_hits=0
Ethernet1/25: PORT is NONE, xoff_hits=0
Ethernet1/26: PORT is NONE, xoff_hits=0
```

```
Ethernet1/27: PORT is NONE, xoff_hits=0
Ethernet1/28: PORT is NONE, xoff_hits=0
Ethernet1/29: PORT is NONE, xoff_hits=0
Ethernet1/30: PORT is NONE, xoff_hits=0
Ethernet1/31: PORT is NONE, xoff_hits=0
Ethernet1/32: PORT is NONE, xoff_hits=0
Ethernet1/33: PORT is NONE, xoff_hits=0
Ethernet1/34: PORT is NONE, xoff_hits=0
Ethernet1/35: PORT is NONE, xoff_hits=0
Ethernet1/36: PORT is NONE, xoff_hits=0
Ethernet1/37: PORT is NONE, xoff_hits=0
Ethernet1/38: PORT is NONE, xoff_hits=0
Ethernet1/39: PORT is NONE, xoff_hits=0
Ethernet1/40: PORT is NONE, xoff_hits=0
Ethernet1/41: PORT is NONE, xoff_hits=0
Ethernet1/42: PORT is NONE, xoff_hits=0
Ethernet1/43: PORT is NONE, xoff_hits=0
Ethernet1/44: PORT is NONE, xoff_hits=0
Ethernet1/45: PORT is NONE, xoff_hits=0
Ethernet1/46: PORT is NONE, xoff_hits=0
Ethernet1/47: PORT is NONE, xoff_hits=0
Ethernet1/48: PORT is NONE, xoff_hits=0
Ethernet1/49: PORT is NONE, xoff_hits=0
Ethernet1/49/2: PORT is NONE, xoff_hits=0
Ethernet1/49/3: PORT is NONE, xoff_hits=0
Ethernet1/49/4: PORT is NONE, xoff_hits=0
Ethernet1/50: PORT is NONE, xoff_hits=0
Ethernet1/50/2: PORT is NONE, xoff_hits=0
Ethernet1/50/3: PORT is NONE, xoff_hits=0
Ethernet1/50/4: PORT is NONE, xoff_hits=0
Ethernet1/51: PORT is NONE, xoff_hits=0
Ethernet1/51/2: PORT is NONE, xoff_hits=0
Ethernet1/51/3: PORT is NONE, xoff_hits=0
Ethernet1/51/4: PORT is NONE, xoff_hits=0
Ethernet1/52: PORT is NONE, xoff_hits=0
Ethernet1/52/2: PORT is NONE, xoff_hits=0
Ethernet1/52/3: PORT is NONE, xoff_hits=0
Ethernet1/52/4: PORT is NONE, xoff_hits=0
Ethernet1/53: PORT is NONE, xoff_hits=0
Ethernet1/53/2: PORT is NONE, xoff_hits=0
Ethernet1/53/3: PORT is NONE, xoff_hits=0
Ethernet1/53/4: PORT is NONE, xoff_hits=0
Ethernet1/54: PORT is NONE, xoff_hits=0
Ethernet1/54/2: PORT is NONE, xoff_hits=0
Ethernet1/54/3: PORT is NONE, xoff_hits=0
Ethernet1/54/4: PORT is NONE, xoff_hits=0

module-1#
```

This example shows syslog messages that are displayed when a pause frame timeout occurs:

```
2021 Jun 25 10:07:41 StArcher-Peer1 %TAHUSD-SLOT1-2-TAHUSD_SYSLOG_CRIT:
   PAUSE-TIMEOUT_BEGIN: Ethernet1/23, PFC pause timeout of 500ms reached for qos_group 1
cos 3 occurrences 1,
   setting port to drop class traffic
2021 Jun 25 10:08:23 StArcher-Peer1 %TAHUSD-SLOT1-2-TAHUSD_SYSLOG_CRIT:
   PAUSE-TIMEOUT_END: Ethernet1/23, PFC pause timeout ended for qos_group 1 cos 3 duration
 40 seconds,
   setting port to transmit class traffic
```

Verifying the FCoE NPV Configuration

To display FCoE/VPC configuration information, perform one of the following:

Command	Purpose
show fcoe	Displays status of Fibre Channel over Ethernet (FCoE) parameters on the switch.
show fcoe database	Displays content of the Fibre Channel over Ethernet (FCoE) database.
show int vfc *vfc-id*	Displays vFC interface information.

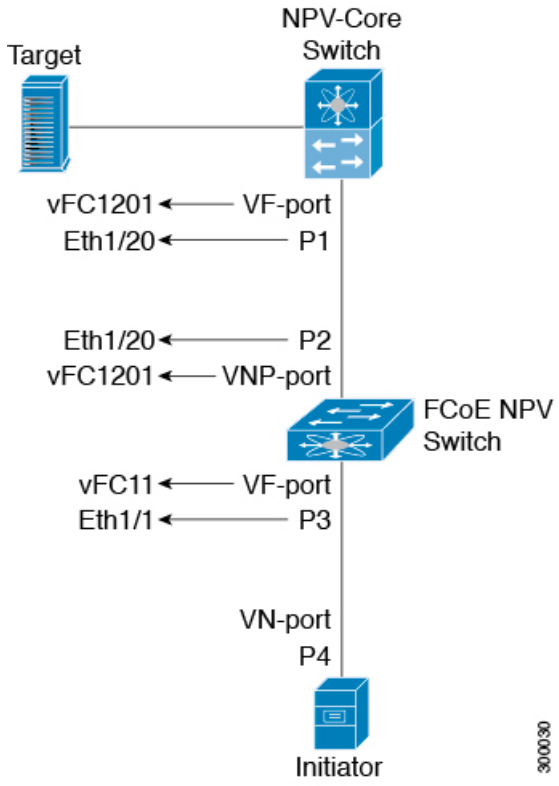
To display NPV configuration information, perform one of the following:

Command	Purpose
show npv status	Displays N Port Virtualization (NPV) current status
show npv traffic-map	Displays N Port Virtualization (NPV) traffic map.
show npv external-interface-usage server-interface *if*	Displays external vFC interfaces (NP interfaces) designated/allocated to the server vFC interface *if* through automatic or static allocation.
show npv external-interface-usage	Displays external vFC interfaces (NP interfaces) designated/allocated to all available server vFC interfaces through automatic or static allocation.
show npv flogi-table interface *if*	Displays host FLOGI table that lists server interface; VSAN; fcid allocated to the initiator connected to the server interface; PWWN and NWWN of initiator; and external interface/gateway on NPV switch designated to the server interface.
show npv flogi-table vsan *vsan*	Displays information about N Port Virtualization (NPV) FLOGI session specific to the VSAN.

Command	Purpose
show npv flogi-table	Displays information about N Port Virtualization (NPV) FLOGI session.
show fcoe-npv issu-impact	Displays information about VNP ports where FKA is disabled.

FCoE NPV Core Switch and FCoE NPV Edge Switch Configuration Example

Figure 1: Configuring FCoE NPV Core Switch and FCoE NPV Edge Switch



- Configure NPV core switch:
  - Enable NPIV

```
npv-core(config)# feature npiv
```

  - Set physical interface mode to trunk

```
npv-core(config)# interface Eth 1/20
npv-core(config)# switchport
npv-core(config)# switchport mode trunk
```

```
npv-core(config)# mtu 9216
npv-core(config)# service-policy type qos input default-fcoe-in-policy
```

Note The steps *switchport, MTU and service-policy* are required only when a Cisco Nexus C93180YC-FX, N9K-C9336C-FX2-E, or N9K-C93360YC-FX2 switches are used as a core switch.

- Set vFC port mode of P1 to VF

```
npv-core(config)# interface vfc1201
npv-core(config)# bind interface Eth1/20
npv-core(config)# switchport mode F
```

- Create VSAN and add vFC into VSAN

```
npv-core(config)# vsan database
npv-core(config-vsan-db)# vsan 100
npv-core(config-vsan-db)# vsan 100 interface vfc1201
```

- Create FCoE VLAN & map it to VSAN

```
npv-core(config)# vlan 100
npv-core(config-vlan)# fcoe vsan 100
```

- Configure FCoE NPV switch:
  - Install FCoE NPV

```
npv(config)# install feature-set fcoe-npv
```

  - Enable FCoE NPV

```
npv(config)# feature-set fcoe-npv
```

  - Create VSAN

```
npv(config)# vsan database
npv(config-vsan-db)# vsan 100
```

  - Create FCoE VLAN and map it to VSAN

```
npv(config)# vlan 100
npv(config-vlan)# fcoe vsan 100
```

  - Set switch side physical interface to trunk mode

```
npv(config)# interface Eth 1/20
npv(config-if)# switchport mode trunk
npv(config-if)# mtu 9216
npv(config-if)# service-policy type qos input default-fcoe-in-policy
```

  - Set vFC port mode of P2 to VNP

```
npv(config)# interface vfc1201
npv(config-if)# switchport mode NP
npv(config-if)# bind interface Eth1/20
```

  - Set server side physical interface to trunk mode

-
```
npv(config)# interface Eth 1/1
npv(config-if)# switchport mode trunk
npv(config-if)# mtu 9216
npv(config-if)# service-policy type qos input default-fcoe-in-policy
```

- Configure vFC port P3 to allow VSAN 100

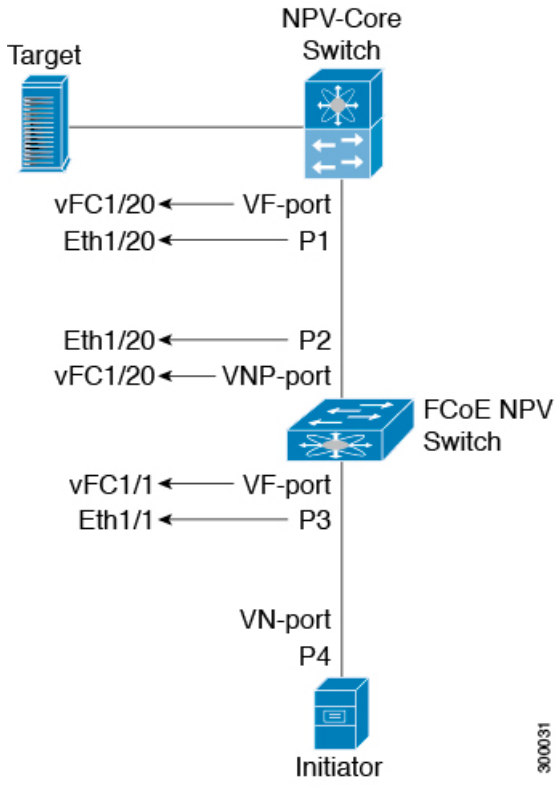
```
npv(config)# interface vfc11
npv(config-if)# switchport trunk allowed vsan 100
npv(config-if)# bind interface Eth1/1
```

- Add both VNP and VF ports into VSAN 100

```
npv(config)# vsan database
npv(config-vsan-db)# vsan 100 interface vfc1201
npv(config-vsan-db)# vsan 100 interface vfc11
```

FCoE NPV Core Switch and FCoE NPV Edge Switch with Implicit vFC Configuration Example

Figure 2: Configuring FCoE NPV Core Switch and FCoE NPV Edge Switch with Implicit vFC



- Configure NPV core switch:
  - Enable NPIV

```
npv-core(config)# feature npiv
```

  - Set physical interface mode to trunk

```
npv-core(config)# interface Eth 1/20
npv-core(config)# switchport
```

```
npv-core(config)# switchport mode trunk
npv-core(config)# mtu 9216
npv-core(config)# service-policy type qos input default-fcoe-in-policy
```

Note The steps *switchport, MTU and service-policy* are required only when a Cisco Nexus C93180YC-FX, N9K-C9336C-FX2-E, or N9K-C93360YC-FX2 switches are used as a core switch.

- Set vFC port mode of P1 to VF (implicit vFC)

```
npv-core(config)# interface vfc 1/20
npv-core(config)# switchport mode F
```

- Create VSAN and add vFC into VSAN

```
npv-core(config)# vsan database
npv-core(config-vsan-db)# vsan 100
npv-core(config-vsan-db)# vsan 100 interface vfc 1/20
```

- Create FCoE VLAN & map it to VSAN

```
npv-core(config)# vlan 100
npv-core(config-vlan)# fcoe vsan 100
```

- Configure FCoE NPV switch:
  - Install FCoE NPV

```
npv(config)# install feature-set fcoe-npv
```

  - Enable FCoE NPV

```
npv(config)# feature-set fcoe-npv
```

  - Create VSAN

```
npv(config)# vsan database
npv(config-vsan-db)# vsan 100
```

  - Create FCoE VLAN and map it to VSAN

```
npv(config)# vlan 100
npv(config-vlan)# fcoe vsan 100
```

  - Set switch side physical interface to trunk mode

```
npv(config)# interface Eth 1/20
npv(config-if)# switchport mode trunk
npv(config-if)# mtu 9216
npv(config-if)# service-policy type qos input default-fcoe-in-policy
```

  - Set vFC port mode of P2 to VNP (implicit vFC)

```
npv(config)# interface vfc 1/20
npv(config-if)# switchport mode NP
```

  - Set server side physical interface to trunk mode

```
npv(config)# interface Eth 1/1
npv(config-if)# switchport mode trunk
npv(config-if)# mtu 9216
npv(config-if)# service-policy type qos input default-fcoe-in-policy
```

- Configure vFC port P3 to allow VSAN 100 (implicit vFC)

```
npv(config)# interface vfc 1/1
npv(config-if)# switchport trunk allowed vsan 100
```

- Add both VNP and VF ports into VSAN 100

```
npv(config)# vsan database
npv(config-vsan-db)# vsan 100 interface vfc 1/20
npv(config-vsan-db)# vsan 100 interface vfc 1/1
```

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(config-if)# sh int vfc 172

vfc172 is trunking (Not all VSANs UP on the trunk)
    Bound interface is Ethernet1/72
    Hardware is Ethernet
    Port WWN is 20:ab:e0:0e:da:4a:5d:9d
    Admin port mode is F, trunk mode is on
    snmp link state traps are enabled
    Port mode is TF
    Port vsan is 200
    Speed is auto
    Trunk vsans (admin allowed and active) (1,10,100,200)
    Trunk vsans (up)                       (200)
    Trunk vsans (isolated)                 ()
    Trunk vsans (initializing)             (1,10,100)
    799 fcoe in packets
    80220 fcoe in octets
    2199 fcoe out packets
    2219828 fcoe out octets
    Interface last changed at Thu Sep 15 08:52:51 2016
```

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

```
switch(config-if)# sh int vfc 132

vfc132 is trunking (Not all VSANs UP on the trunk)
    Bound MAC is 000e.1e1b.c1c9
    Hardware is Ethernet
    Port WWN is 20:83:00:2a:10:7a:89:bf
    Admin port mode is F, trunk mode is on
    snmp link state traps are enabled
```

```
    Port mode is TF
    Port vsan is 2101
    Speed is auto
    Trunk vsans (admin allowed and active) (1,2001-2003,2101-2103)
    Trunk vsans (up)                       (2101)
    Trunk vsans (isolated)                 ()
    Trunk vsans (initializing)             (1,2001-2003,2102-2103)
    Interface last changed at Wed Sep 14 12:14:29 2016
```

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:

SUMMARY STEPS

1. Enable the associated VLAN and map the VLAN to a VSAN.
2. Configure the VLAN on a physical Ethernet interface.
3. Create a virtual Fibre Channel interface and bind it to a physical Ethernet interface.
4. Associate the virtual Fibre Channel interface to the VSAN.
5. (Optional) Display membership information for the VSAN.
6. (Optional) Display the interface information for the virtual Fibre Channel interface.

DETAILED STEPS

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.

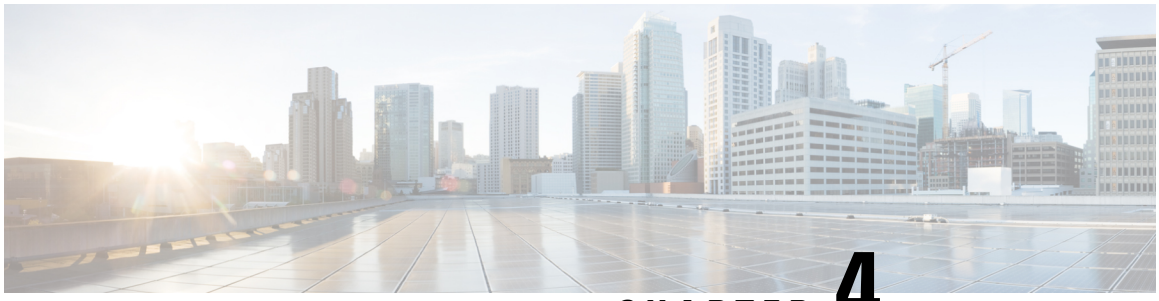
```
switch# show interface vfc 4

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
```

SAN Boot with vPC

Cisco Nexus 9000 Series devices support the SAN boot of initiators on Link Aggregation Control Protocol (LACP) based vPC. This limitation is specific to LACP-based port channels. The host-facing vFC interfaces are bound to port channel members instead of the port channel itself. This binding ensures that the host-side vFC comes up during a SAN boot as soon as the link on the CNA/Host Bus Adapter (HBA) comes up, without relying on the LACP-based port channel to form first.

Note Cisco Nexus 9000 Series devices support the SAN boot of channel mode on also.

Note LACP suspend-individual should be removed from the port-channel, otherwise the physical interface will be suspended when LACP BPDU is not received from the host.



CHAPTER 4

FCoE Over FEX

Overview

The Fibre Channel over Ethernet (FCoE) over Fabric Extenders (FEX) feature allows Fibre Channel traffic to be carried on a FEX port. The FEX is connected to a Cisco Nexus 9000 device that is in FCoE NPV mode through a Fabric Port Channel (FPC). FCoE over FEX enables the provisioning of FCoE on host connections.

For more information about FEX, see the *Cisco Nexus 2000 Series NX-OS Fabric Extender Configuration Guide for Cisco Nexus 9000 Series Switches*.

FCoE Over FEX with vPC

FCoE over FEX with virtual Port Channel (vPC) allows Fibre Channel traffic to be carried over a FEX using a virtual Port Channel (vPC).

LAN Shutdown

The LAN shutdown feature detects the capability of the FCoE host to support Data Center Bridging (DCBX). DCBX allows the switch to send the LAN Logical Link status (LLS) messages in a type-length-value (TLV) format. The LAN shutdown feature enables bring up and bring down of LAN links on a unified link carrying both FCoE and LAN traffic. When you use the **shutdown lan** command, only the LAN traffic stops while the FCoE traffic continues.

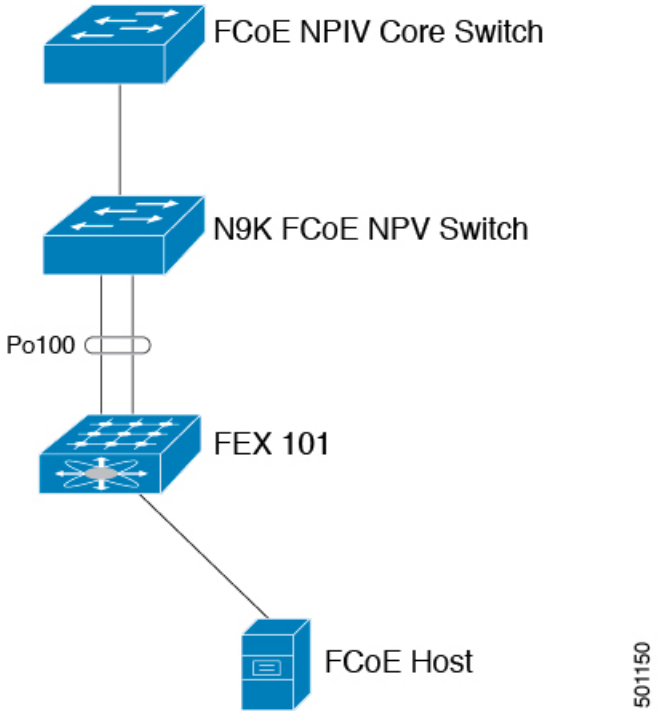
The **shutdown lan** command is supported for FEX HIF ports and port-channels.

FCoE Over FEX Topologies

FCoE over FEX is supported in the following topologies:

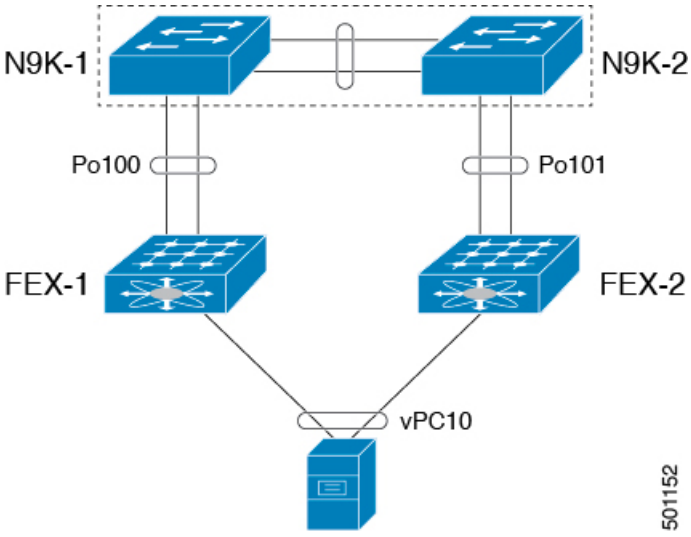
Straight Through FEX with Host Topology

The straight through FEX with host topology is supported with Cisco NX-OS Release 9.3(3) and later.



Straight Through FEX with Host VPC Topology

The straight through FEX with host VPC topology is supported with Cisco NX-OS Release 9.3(3) and later.



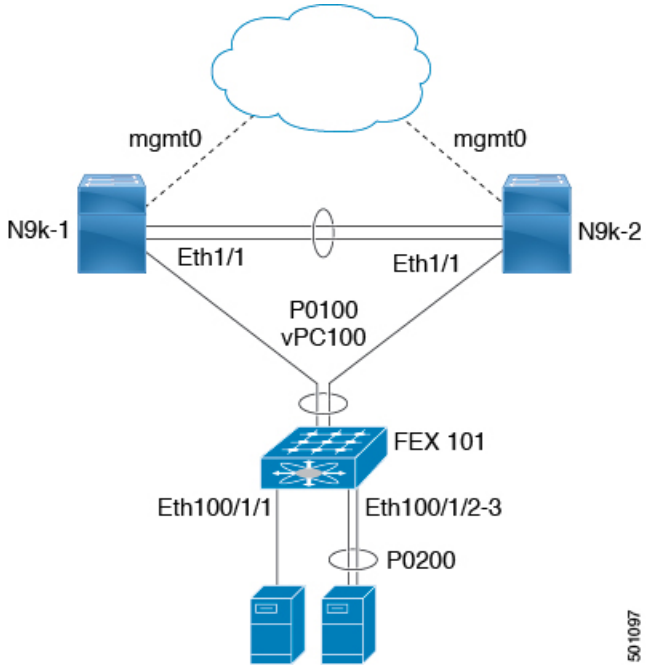
Dual-Homed FEX Topology (Active/Active FEX Topology)

The dual-homed FEX topology is supported with Cisco NX-OS Release 9.3(3) and later with Cisco Nexus 9300-EX and 9300-FX Series switches.

The following topology shows that each FEX is dual-homed with two Cisco Nexus 9000 Series switches. The FEX-fabric interfaces for each FEX are configured as a vPC on both peer switches. The host interfaces on the FEX appear on both peer switches.

Note The host interfaces configuration should be the same on both switches.

Figure 3: Dual-Homed FEX Topology



Operational vPC is mandatory for the dual-homed FEX topology. In the Dual-Homed FEX Active/Active Topology, the vPC is already operational. FEX 101 is dual-homed to both parent switches: N9k-1 and N9k-2 on FEX-fabric interfaces Ethernet 1/1.

Note Only the following support an active-active FEX topology:

- N2K-C2232PP
- N2K-C2348UPQ
- NB22IBM
- NB22HP

Note A port channel within the same FEX is supported on Cisco Nexus 2200 Series Fabric Extenders.

Guidelines and Limitations for FCoE Over FEX

- In FEX AA configuration, if FCoE pinning is done on the secondary switch, the FCoE traffic is disrupted when the peer link is flapped.
- FCoE host connected to the FEX can login via both FC and FCoE NP uplinks in N9K-C93180YC-FX.
- If a traffic map is configured for HIF ports for Cisco Nexus 93180YC-FX switches, make sure that all the hosts in the same FEX are mapped to the same NP link. For other Cisco Nexus (older) switches, traffic maps for the FEX HIF ports work the same way as before.
- Any or all VSANs that are configured as port VSAN of HIFs should be allowed on all the NP links (external interfaces).
- N9K-C93180LC-EX supports FCoE over FEX only on the N2K-C2348UPQ. Other FEX models are not supported with this device.
- FEX HIF ports only support vFC in F mode. NP mode vFCs are not supported over FEX HIF ports.
- N9K-C93180YC-FX3 switch supports FEX when used with the N9K-C93180YC-FX parent switch only. Ensure that you follow the below guidelines while configuring FEX on the switch:
  - Make sure to disable auto-negotiation when using 40G or 100G FEX NIF uplink ports.
  - For a 10G connection, make sure to connect to uplink ports 49 or higher.
- A Fabric Port Channel cannot exceed a maximum of eight member ports.
- 4q policies are supported on FCoE over FEX.
- 8q policies are not supported on FCoE over FEX.
- FC ports are not supported on FEX.

Note For information about scalability, see the Cisco Nexus 9000 Series NX-OS Verified Scalability Guide.

Configuring FCoE Over FEX

Configuring Straight Through FEX with Host

Note Considerations for FEX Fabric Port Channel (FPC):

- Priority flow control (PFC) requires to be enabled explicitly for Pause to work.

```
interface "port-channel"
priority-flow-control mode on
```

- If the switch is connected to a host that does not have DCBX support, PFC needs to be enabled explicitly on HIF interfaces.

```
interface "hif interface"
priority-flow-control mode on
```

- The LLDP feature should be enabled on the switch with the **feature lldp** command.
- First attach the interfaces to the fex-fabric port-channel and then enable the priority-flow-control mode with the **priority-flow-control mode on** command.
- The **fcoe enable-fex** command is required for the FCoE host logins to work properly when connected to the FEX HIF ports. This command is supported and required only on the following Cisco Nexus 9000 series switches such as N9K-C9332PQ, N9K-C9372PX-E, N9K-C9372PX, N9K-C9396PX, N9K-X9464PX line cards, and N9K-X9564PX line cards.

Before you begin

- Ensure configuration of FEX. For more information, see the *Cisco Nexus 2000 Series NX-OS Fabric Extender Configuration Guide for Cisco Nexus 9000 Series Switches*.
- Ensure the configuration of FCoE NPV. For more information, see the section: *Configuring FCoE NPV*.
- You must apply the QoS policy at the system (global) level and to the Host Interfaces (HIF's) to honor the pause frames in the FCoE traffic.

```
switch(config)# system qos
switch(config-sys-qos)# service-policy type queuing input policy-name
switch(config-sys-qos)# service-policy type queuing output policy-name
switch(config-sys-qos)# service-policy type network-qos policy-name
switch(config-sys-qos)# service-policy type qos input policy-name
```

SUMMARY STEPS

1. **configure terminal**
2. **interface port-channel** <*port_num*>
3. **switchport**

4. **switchport mode fex-fabric**
5. **fex associate** *<fex_id>*
6. **mtu 9216**
7. **no shutdown**
8. **exit**
9. **interface ethernet** *slot/port*
10. **switchport**
11. **switchport mode fex-fabric**
12. **fex associate** *<fex_id>*
13. **mtu 9216**
14. **channel-group** *<port_number>*
15. **no shutdown**
16. **exit**
17. **interface ethernet** *chassis_id/slot/port_number*
18. **switchport mode trunk**
19. **service-policy type qos input fcoe**
20. **no shutdown**
21. **exit**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**interface port-channel** *<port_num>* **Example:** `switch(config)#interface port-channel 101`	Creates port-channel and enters the interface configuration mode.
Step 3	**switchport** **Example:** `switch(config-if)#switchport`	Sets Layer 2 switching port.
Step 4	**switchport mode fex-fabric** **Example:** `switch(config-if)#switchport mode fex-fabric`	Sets the interface type to be a uplink port for a Fabric extender (FEX).
Step 5	**fex associate** *<fex_id>* **Example:** `switch(config-if)#fex associate 101`	Associates a Fabric Extender (FEX) to a Fabric interface.

	Command or Action	Purpose
Step 6	**mtu 9216** **Example:** `switch(config-if)#mtu 9216`	Configures the MTU value to that of jumbo frames to carry FCoE traffic through the NIF ports.
Step 7	**no shutdown** **Example:** `switch(config-if)#no shutdown`	Sets the port-channel to up (administratively).
Step 8	**exit** **Example:** `switch(config-if)#exit`	Exits the interface configuration mode.
Step 9	**interface ethernet** *slot*/*port* **Example:** `switch(config)#interface Ethernet1/1`	Enters interface configuration mode.
Step 10	**switchport** **Example:** `switch(config-if)#switchport`	Sets Layer 2 switching port.
Step 11	**switchport mode fex-fabric** **Example:** `switch(config-if)#switchport mode fex-fabric`	Sets the interface type to be an uplink port for a fabric extender (FEX).
Step 12	**fex associate** <*fex_id*> **Example:** `switch(config-if)#fex associate 101`	Associates a Fabric Extender (FEX) to a Fabric interface.
Step 13	**mtu 9216** **Example:** `switch(config-if)#mtu 9216`	Configures the MTU value to that of jumbo frames to carry FCoE traffic through the NIF ports.
Step 14	**channel-group** <*port_number*> **Example:** `switch(config-if)channel-group 101`	Sets the Fabric interface a member of the Fabric port channel.
Step 15	**no shutdown** **Example:**	Sets the port-channel to up (administratively).

	Command or Action	Purpose
	`switch(config-if)#no shutdown`	
Step 16	**exit** **Example:** `switch(config-if)#exit`	Exits the interface configuration mode.
Step 17	**interface ethernet** *chassis_id*/*slot*/*port_number* **Example:** `switch(config)interface ethernet 101/1/1`	Configures a FEX satellite interface or HIF (host interface) port and enters interface configuration mode.
Step 18	**switchport mode trunk** **Example:** `switch(config-if)#switchport mode trunk`	Sets interface type to be a trunk port.
Step 19	**service-policy type qos input fcoe** **Example:** `switch(config-if)# service-policy type qos input fcoe`	Sets the QoS policy on the HIF port channel to a no drop policy. **Note** For more information about queuing policy configurations, see the section: *Configuring QoS for no-drop support*.
Step 20	**no shutdown** **Example:** `switch(config-if)#no shutdown`	Sets the port-channel to up (administratively).
Step 21	**exit** **Example:** `switch(config-if)#exit`	Exits the interface configuration mode.

Example

The following is an example of FEX bring up in straight through mode with a host.

```
install feature-set fex
feature-set fex

fex 101
  pinning max-links 1
  description "2232PP-1"

interface port-channel101
  switchport
  switchport mode fex-fabric
  fex associate 101
```

```
  mtu 9216

interface Ethernet1/1
  switchport
  switchport mode fex-fabric
  fex associate 101
  mtu 9216
  channel-group 101
  no shutdown

interface Ethernet101/1/1
  switchport mode trunk
  service-policy type qos input fcoe-qo-policy
  no shutdown
```

Binding vFC to FEX Interface Explicitly

SUMMARY STEPS

1. **configure terminal**
2. **interface vfc** < *vfc-id*>
3. **bind interface ethernet** [*chassid-id*/*slot*/*port*]
4. **no shutdown**
5. **end**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**interface vfc** < *vfc-id*> **Example:** `N9k(config)# interface vfc 1`	Creates a virtual Fibre Channel interface and enters the interface configuration mode. The chassis-id range is from 101 to 199.
Step 3	**bind interface ethernet** [*chassid-id*/*slot*/*port*] **Example:** `N9k(config-if)# bind interface ethernet101/1/1`	Explicitly binds the virtual fibre channel interface to the specified interface. Use the **no** form of the command to unbind the interface. Chassis id range is from 101 to 199.
Step 4	**no shutdown** **Example:** `switch(config-if)#no shutdown`	Brings up the interface (administratively).

	Command or Action	Purpose
Step 5	**end** **Example:** `N9k(config-if)#end`	Returns to exec mode.

Example

```
interface vfc1
  bind interface ethernet 101/1/1
  switchport trunk mode on
  no shutdown
```

Binding VFC to FEX Interface Implicitly

SUMMARY STEPS

1. **configure terminal**
2. **interface vfc** < *chassis-id*>/<*slot*>/<*port*>
3. **no shutdown**
4. **end**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**interface vfc** < *chassis-id*>/<*slot*>/<*port*> **Example:** `switch(config)# interface vfc 101/1/1`	Creates a virtual fibre Channel Interface and enters interface configuration mode; implicitly binding it to the underlying Ethernet interface (ethernet chassid-id/slot/port). The chassis-id range is from 101 to 199.
Step 3	**no shutdown** **Example:** `switch(config-if)#no shutdown`	Brings up the interface (administratively).
Step 4	**end** **Example:** `switch(config-if)#end`	Returns to exec mode.

Example

```
interface vfc101/1/1
  switchport trunk mode on
  no shutdown
```

Binding vFC to MAC Address

A MAC address bound vFC can also be created for a FEX host interface (HIF) port.

Note A MAC bound vFC can be configured to a host sitting behind a FIP Snooping Bridge (FSB).

When both MAC bound vFC and port-bound vFC are configured for the same interface, the port-bound vFC takes precedence.

As a best practice, you should have either a MAC bound vFC or a port- bound vFC for a physical Ethernet port or a port-channel. However, you cannot have both.

SUMMARY STEPS

1. **configure terminal**
2. **interface vfc** <*number*>
3. **bind mac-address** <*mac-address*>

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**interface vfc** <*number*>	Creates a virtual fibre Channel Interface.
Step 3	**bind mac-address** <*mac-address*>	Binds the MAC address.

Example

The following 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
```

Configuring Straight Through FEX with Host vPC

Before you begin

- Ensure configuration of vPC between two Cisco Nexus 9000 Series switches. For more information, see the *Cisco Nexus 9000 Series NX-OS Interfaces Configuration Guide*.
- Ensure configuration of FEX. For more information, see the *Cisco Nexus 2000 Series NX-OS Fabric Extender Configuration Guide for Cisco Nexus 9000 Series Switches*.
- Ensure the configuration of FCoE NPV. For more information, see the section: *Configuring FCoE NPV*.

SUMMARY STEPS

1. **configure terminal**
2. **interface port-channel** <*port_num*>
3. **switchport**
4. **switchport mode fex-fabric**
5. **fex associate** <*fex_id*>
6. **mtu 9216**
7. **no shutdown**
8. **exit**
9. **interface ethernet** *slot*/*port*
10. **switchport**
11. **switchport mode fex-fabric**
12. **fex associate** <*fex_id*>
13. **mtu 9216**
14. **channel-group** <*port_number*>
15. **no shutdown**
16. **exit**
17. **interface ethernet** *chassis_id*/*slot*/*port_number*
18. **switchport mode trunk**
19. **channel group** <*host_port_num*>
20. **no shutdown**
21. **exit**
22. **interface port-channel** <*host_port_number*>
23. **switchport**
24. **switchport mode trunk**
25. **service-policy type qos input fcoe**
26. **vpc 3**
27. **no shutdown**
28. **exit**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**interface port-channel** <*port_num*> **Example:** `switch(config)#interface port-channel 101`	Creates port-channel and enters the interface configuration mode.
Step 3	**switchport** **Example:** `switch(config-if)#switchport`	Sets Layer 2 switching port.
Step 4	**switchport mode fex-fabric** **Example:** `switch(config-if)#switchport mode fex-fabric`	Sets the interface type to be a uplink port for a Fabric extender (FEX).
Step 5	**fex associate** <*fex_id*> **Example:** `switch(config-if)#fex associate 101`	Associates a Fabric Extender (FEX) to a Fabric interface.
Step 6	**mtu 9216** **Example:** `switch(config-if)#mtu 9216`	Configures the MTU value to that of jumbo frames to carry FCoE traffic through the NIF ports.
Step 7	**no shutdown** **Example:** `switch(config-if)#no shutdown`	Sets the port-channel to up (administratively).
Step 8	**exit** **Example:** `switch(config-if)#exit`	Exits the interface configuration mode.
Step 9	**interface ethernet** *slot*/*port* **Example:** `switch(config)#interface Ethernet1/1`	Enters the interface configuration mode.

	Command or Action	Purpose
Step 10	**switchport** **Example:** `switch(config-if)#switchport`	Sets Layer 2 switching port.
Step 11	**switchport mode fex-fabric** **Example:** `switch(config-if)#switchport mode fex-fabric`	Sets the interface type to be an uplink port for a fabric extender (FEX).
Step 12	**fex associate** <*fex_id*> **Example:** `switch(config-if)#fex associate 101`	Associates a Fabric Extender (FEX) to a Fabric interface.
Step 13	**mtu 9216** **Example:** `switch(config-if)#mtu 9216`	Configures the MTU value to that of jumbo frames to carry FCoE traffic through the NIF ports.
Step 14	**channel-group** <*port_number*> **Example:** `switch(config-if)channel-group 101`	Sets the Fabric interface a member of the Fabric port channel.
Step 15	**no shutdown** **Example:** `switch(config-if)#no shutdown`	Sets the port-channel to up (administratively).
Step 16	**exit** **Example:** `switch(config-if)#exit`	Exits the interface configuration mode.
Step 17	**interface ethernet** *chassis_id*/*slot*/*port_number* **Example:** `switch(config)interface ethernet 101/1/1`	Configures a FEX satellite interface or HIF (host interface) port and enters interface configuration mode.
Step 18	**switchport mode trunk** **Example:** `switch(config-if)#switchport mode trunk`	Sets interface type to be a trunk port.
Step 19	**channel group** <*host_port_num*> **Example:**	Makes the HIF port a member of a port-channel

	Command or Action	Purpose
	`switch(config-if)# channel group 1`	
Step 20	**no shutdown** **Example:** `switch(config-if)#no shutdown`	Sets the port-channel to up (administratively).
Step 21	**exit** **Example:** `switch(config-if)#exit`	Exits the interface configuration mode.
Step 22	**interface port-channel** <*host_port_number*> **Example:** `switch(config)#interface port-channel 1`	Creates a HIF port-channel.
Step 23	**switchport** **Example:** `switch(config-if)#switchport`	Sets Layer 2 switching port.
Step 24	**switchport mode trunk** **Example:** `switch(config-if)#switchport mode trunk`	Sets the interface to be a trunk port.
Step 25	**service-policy type qos input fcoe** **Example:** `switch(config-if)# service-policy type qos input fcoe`	Sets the QoS policy on the HIF port channel to a no drop policy. **Note** For more information about queuing policy configurations, see the section: *Configuring QoS for no-drop support*.
Step 26	**vpc 3**	Configures VPC on the HIF port-channel. The VPC id on both the peers should be same for this Host VPC.
Step 27	**no shutdown** **Example:** `switch(config-if)#no shutdown`	Sets the port-channel to up (administratively).
Step 28	**exit** **Example:** `switch(config-if)#exit`	Exits the interface configuration mode.

Example

The following is an example of FEX bring up in straight through mode with a host vPC.

- Configuration on Peer-1

```
install feature-set fex
feature-set fex

fex 101
  pinning max-links 1
  description "2232PP-1"

interface port-channel101
  switchport
  switchport mode fex-fabric
  fex associate 101
  mtu 9216

interface Ethernet1/1
  switchport
  switchport mode fex-fabric
  fex associate 101
  mtu 9216
  channel-group 101
  no shutdown

interface Ethernet101/1/1
  switchport mode trunk
  channel-group 1
  no shutdown

interface port-channel1
  switchport
  switchport mode trunk
  service-policy type qos input fcoe
  vpc 3

interface vfc-po1      /*** Implicit binding with VFC bound to port-channel ***/
  bind interface port-channel1
  switchport trunk mode on
  no shutdown

interface vfc101/1/1   /*** Implicit binding with VFC bound to member port ***/
  bind interface ethernet101/1/1
  switchport trunk mode on
  no shutdown
```

- Configuration on Peer-2

```
install feature-set fex
feature-set fex

fex 102
  pinning max-links 1
  description "2348UPQ-2"

interface port-channel102
  switchport
  switchport mode fex-fabric
```

```
  fex associate 102
  mtu 9216

interface Ethernet102/1/1
  switchport mode trunk
  channel-group 1
  no shutdown

interface port-channel1
  switchport
  switchport mode trunk
  service-policy type qos input fcoe
  vpc 3

interface vfc1   /*** Explicit binding with VFC bound to port-channel ***/
  bind interface port-channel1
  switchport trunk mode on
  no shutdown

interface vfc2   /*** Explicit binding with VFC bound to member port ***/
  bind interface ethernet102/1/1
  switchport trunk mode on
  no shutdown
```

Note The vFC binding should be either to the port-channel or to the member host interface.

Configuring Dual-Homed FEX

SUMMARY STEPS

1. **configure terminal**
2. **fex fex-chassis_ID**
3. **fcoe**
4. **interface port-channel** <*port_num*>
5. **switchport**
6. **switchport mode fex-fabric**
7. **fex associate** <*fex_id*>
8. **mtu 9216**
9. **vpc 1**
10. **no shutdown**
11. **exit**
12. **interface ethernet** *slot*/*port*
13. **switchport**
14. **switchport mode trunk**
15. **service-policy type qos input fcoe-qo-policy**
16. **no shutdown**
17. **exit**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**fex fex-chassis_ID**
	Example:
	`switch# fex 101`
	`switch(config)#`
		Enters the configuration mode for the specified FEX. The range for fex-chassis_ID is from 101 to 199.

Step 3	**fcoe**
	Example:
	`switch# fcoe`
	`switch(config)#`
		Configures the FEX to send FCoE traffic only to this switch.
		Note Cisco recommends that FCOE pinning is configured on the primary vPC switch in case of dual homed FEX.

Step 4	**interface port-channel** <*port_num*> **Example:** `switch(config)#interface port-channel170`	Creates port-channel and enters the interface configuration mode.
Step 5	**switchport** **Example:** `switch(config-if)#switchport`	Sets Layer 2 switching port.
Step 6	**switchport mode fex-fabric** **Example:** `switch(config-if)#switchport mode fex-fabric`	Sets the interface type to be a uplink port for a Fabric extender (FEX).
Step 7	**fex associate** <*fex_id*> **Example:** `switch(config-if)#fex associate 170`	Associates a Fabric Extender (FEX) to a Fabric interface.
Step 8	**mtu 9216** **Example:** `switch(config-if)#mtu 9216`	Configures the MTU value to that of jumbo frames to carry FCoE traffic through the NIF ports.
Step 9	**vpc 1**	Configures VPC on the HIF port-channel.

	Command or Action	Purpose
Step 10	**no shutdown** **Example:** `switch(config-if)#no shutdown`	Sets the port-channel to up (administratively).
Step 11	**exit** **Example:** `switch(config-if)#exit`	Exits the interface configuration mode.
Step 12	**interface ethernet** *slot*/*port* **Example:** `switch(config)#interface Ethernet170/1/18`	Enters interface configuration mode.
Step 13	**switchport** **Example:** `switch(config-if)#switchport`	Sets Layer 2 switching port.
Step 14	**switchport mode trunk** **Example:** `switch(config-if)#switchport mode trunk`	Sets the interface to be a trunk port.
Step 15	**service-policy type qos input fcoe-qo-policy** **Example:** `switch(config-if)# service-policy type qos input fcoe`	Sets the QoS policy on the HIF port channel to a no drop policy. **Note** For more information about queuing policy configurations, see the section: *Configuring QoS for no-drop support*.
Step 16	**no shutdown** **Example:** `switch(config-if)#no shutdown`	Sets the port-channel to up (administratively).
Step 17	**exit** **Example:** `switch(config-if)#exit`	Exits the interface configuration mode. **Note** The same configuration should be performed on the other side for the Active-Active FEX to be up on both the sides.

Example

Note Cisco recommends that FCOE pinning is configured on the primary vPC switch in case of dual homed FEX.

- Peer 1

```
fex 170
  pinning max-links 1
  description "2232PP-3 AA"
  fcoe

interface port-channel170
  switchport
  switchport mode fex-fabric
  fex associate 170
  mtu 9216
  vpc 1

interface Ethernet170/1/18
  switchport mode trunk
  service-policy type qos input fcoe-qo-policy
  no shutdown

interface vfc1718     /*** Explicit binding ***/
  bind interface Ethernet170/1/18
  switchport trunk mode on
  no shutdown

interface vfc170/1/18 /*** Implicit binding ***/
  bind interface Ethernet170/1/18
  switchport trunk mode on
  no shutdown
```

- Peer 2

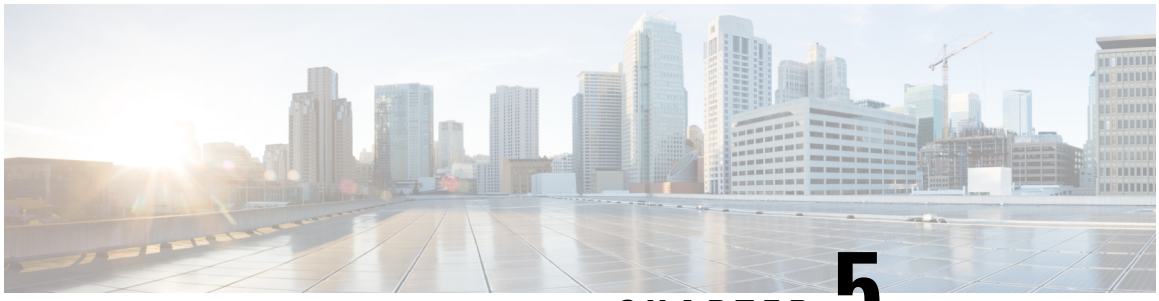
```
fex 170
  pinning max-links 1
  description "2232PP-3 AA"

interface port-channel170
  switchport
  switchport mode fex-fabric
  fex associate 170
  mtu 9216
  vpc 1

interface Ethernet170/1/18
  switchport mode trunk
  service-policy type qos input fcoe-qo-policy
  no shutdown
```

Note

- HOST vPC is not supported with dual-homed FEX.
- The vFC binding must be either to port-channel or to member host interface. vFC cannot be bound to port channel if it has more than one member. vFC cannot be bound to host interface if it is part of a multi member port channel.

Configuring FC NPV

For more information on configuring FC NPV, see Cisco Nexus 9000 Series NX-OS FC NPV Configuration Guide.



CHAPTER 5

Configuring FC NPV

Supported Hardware

FC NPV is supported on N9K-C93180YC-FX, N9K-C9336C-FX2-E, and N9K-C93360YC-FX2 switches.

N9K-C93180YC-FX and N9K-C93360YC-FX2 supports only the following FC SFPs:

- DS-SFP-FC8G-SW
- DS-SFP-FC16G-SW
- DS-SFP-FC32G-SW

N9K-C9336C-FX2-E supports only the following FC SFP:

- DS-SFP-4x32G-SW

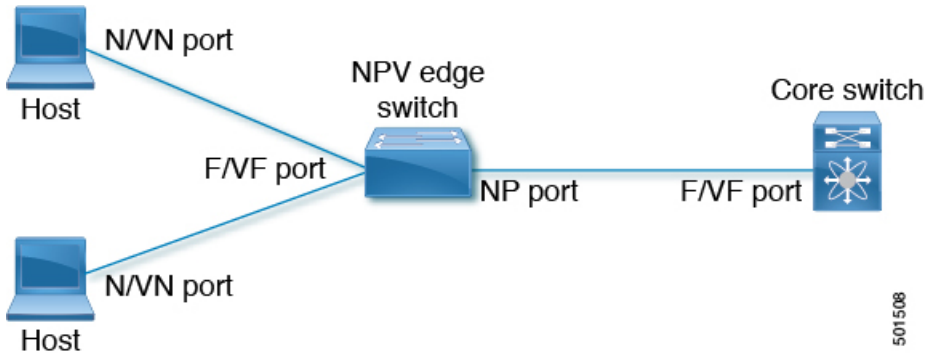
FC NPV Overview

A switch is in NPV mode after enabling NPV. NPV mode applies to an entire switch. All end devices connected to a switch that are in NPV mode must log in as an N port to use this feature (loop-attached devices are not

supported). All links from the edge switches (in NPV mode) to the NPV core switches are established as NP ports (not E ports), which are used for typical inter-switch links. NPIV is used by the switches in NPV mode to log in to multiple end devices that share a link to the NPV core switch.

The following figure shows an interface-level view of an FC NPV configuration.

Figure 4: FC NPV Interface Configuration



FC NPV Benefits

FC NPV provides the following:

- Increased number of hosts that connect to the fabric without adding domain IDs in the fabric
- Connection of FC and FCoE hosts and targets to SAN fabrics using FC interfaces
- Automatic traffic mapping
- Static traffic mapping

FC NPV Mode

In FC NPV mode, the edge switch relays all traffic to the core switch and shares the domain ID of the core switch.

FC NPV is enabled by installing and enabling **feature-set fcoe-npv**. You cannot configure FC NPV mode on a per-interface basis. FC NPV mode applies to the entire switch.

Server Interfaces

- In Cisco Nexus 9000 Series switches, server interfaces can be FC or vFC interfaces.
- Server interfaces are F ports on the edge switch that connect to the servers. A server interface may support multiple end devices by enabling the N port identifier virtualization (NPIV) feature. NPIV provides a means to assign multiple FC IDs to a single N port, which allows the server to assign unique FC IDs to different applications.

Note To use NPIV, enable the NPIV feature and reinitialize the server interfaces that will support multiple devices.

- FC server interfaces should be in trunk mode off. Trunk mode on is not supported.
- vFC server interfaces should be in trunk mode on.
- Server interfaces are automatically distributed among the NP uplinks to the core switch. All of the end devices connected to a server interface are mapped to the same NP uplink.
- When you connect a 16G host adapter to a 32G SFP port on Cisco Nexus 93360YC-FX and/or 93360YC-FX2 switches, the link may not come up when the speed is configured as auto speed. Or sometimes, it defaults to 8G speed. Then, to use 16G speed, you must manually configure the port using the command **switchport speed 16000**.
- 8G speed is not supported for server and target interfaces.

NP Uplinks

- In Cisco Nexus 9000 Series switches, NP uplink interfaces can be native Fibre Channel interfaces, virtual fiber channel interfaces, SAN port channel interfaces, or virtual ethernet port-channel interfaces.
- All interfaces from the edge switch to the core switch are configured as proxy N ports (NP ports).
- An NP uplink is a connection from an NP port on the edge switch to an F port on the core switch. When an NP uplink is established, the edge switch sends a fabric login message (FLOGI) to the core switch then (if the FLOGI is successful) registers itself with the name server on the core switch. Subsequent FLOGIs from end devices connected to this NP uplink are forwarded as-is to the core switch. Subsequent FLOGIs from the same VSAN are forwarded as fdisc.

Note In the switch CLI configuration commands and output displays, NP uplinks are called External Interfaces.

- The default speed of NP links is set to auto.
- The features below must be enabled on the core switch:
  - **feature npiv**
  - **feature fport-channel-trunk**
- If the FC uplink speed is 8G, the fill pattern should be configured as IDLE on the core switch.

Note Following is an example of configuring IDLE fill pattern on a Cisco MDS switch:

```
Switch(config)# int fc2/3
Switch(config)# switchport fill-pattern IDLE speed 8000
Switch(config)# sh run int fc2/3


  interface fc2/3
  switchport speed 8000
  switchport mode NP
  switchport fill-pattern IDLE speed 8000
  no shutdown
```

Note

- To enable trunking and for the FLOGI from NP uplink of Cisco Nexus 9000 Series switches to be successful on the core switch, both the core and the Cisco Nexus 9000 Series switch should be configured with the OUI of each other.

  Configure the OUI on the core and Nexus 9000 switch *only* if the OUI value is not registered by default on either of them.

  The OUI is found and configured as follows:

```
N9K(config-if)# show wwn switch
    Switch WWN is 20:00:2c:d0:2d:50:ea:64
    N9K(config-if)#

On the core, we see the output below if the OUI (0x2cd02d) is already registered.
    MDS9710(config-if)# sh wwn oui | i 2cd02d
    0x2cd02d   Cisco               Default
    MDS9710(config-if) #
If the OUI is not registered with the core, configure it manually.
    MDS9710(config-if)# wwn oui 0x2cd02d
```

- Beginning with Cisco NX-OS Release 7.3(0)D1(1), the OUI is configurable on a Cisco MDS 9700 Series core switches.
- If the uplinks to core switch are FCoE enabled, then the FKA advertisement period is taken from configured value on the core switch. If the uplinks to core switch are FC enabled, FKA a period is taken from configured value on local NPV switch.

Note The following example shows the FCoE uplink. Because the switch has an FCoE link, the value is taken from FCF:

```
switch(config)# sh run fcoe_mgr | i i fka
fcoe fka-adv-period 12

switch(config)# sh fcoe

FCF details for interface vfc-po142
        FCF-MAC is 54:7f:ee:ec:71:84
        FC-MAP is 0e:fc:00
        FCF Priority is 128
        FKA Advertisement period for FCF is 8 seconds <<<<<
```

The following example shows the FC uplink:

```
switch(config)# sh run | i i fka
fcoe fka-adv-period 10

switch(config)# sh fcoe
FCF details for interface san-port-channel29
        FCF-MAC is 2c:d0:2d:50:e4:29
        FC-MAP is 0e:fc:00
        FCF Priority is 129
        FKA Advertisement period for FCF is 10 seconds
```

SAN Port Channels

About SAN Port Channels

- A SAN port channel is a logical interface that combines a set of FC interfaces connected to the same fibre channel node and operates as one link.
- SAN port channels support bandwidth utilization and availability.
- SAN port channels on Cisco Nexus 9000 Series switches are mainly used to connect to MDS cores and to provide optimal bandwidth utilization and transparent failover between the uplinks of a VSAN.

Configuring SAN Port Channels

When you configure a SAN port channel, it gets created with default values . You can modify all the the default values, except the channel mode. You must connect each switch to same number of interfaces on either side of a SAN port channel. Otherwise, you see a SAN port channel error.

SAN Port Channel Guidelines and Limitations

- The number of SAN port channels and vFC port channels, together, can be only 8 on the Cisco Nexus 9000 Series switch.
- The maximum number of FC interfaces that can be combined into a SAN port channel is limited to 16.
- The default channel mode on Cisco Nexus 9000 Series switches for SAN port channels is **active**; this cannot be changed.
- Operating speed and member addition to san-po limitation on N9K-C9336C-FX2-E are available in FC NPV Guidelines and Limitations, on page 79.

Creating a SAN Port Channel

This section explains how to create a SAN port channel.

Step 1 switch# **configure terminal**

Enters the global configuration mode.

Step 2 switch(config)# **interface san-port-channel** *channel-number*

Creates the specified SAN port channel using the default mode (on). The SAN port channel number is in the range of 1 to 256

The following example shows the SAN port channel creation:

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

About SAN Port Channel Modes

A SAN port channel is configured with channel mode active by default. When active, the member ports initiate port-channel-protocol negotiation with the peer port(s) regardless of the channel-group mode of the peer port. If the peer port, while configured in a channel group, does not support the port-channel protocol, or responds with a nonnegotiable status, the port channel will be disabled. The active port -channel mode allows automatic recovery without explicitly enabling and disabling the port-channel-member ports at either end.

About Deleting SAN Port Channels

When you delete the SAN port channel, the corresponding channel membership is also deleted.

If you delete the SAN port channel for one port, then the individual ports within the deleted SAN port channel retain the compatibility parameter settings (speed, mode, port VSAN, allowed VSAN, and port security). You can explicitly change those settings as required.

Deleting SAN Port Channels

This section explains how to delete a SAN port channel.

Step 1 switch# **configure terminal**

Enters global configuration mode.

Step 2 switch(config)#**no interface san-port-channel** *channel-number*

Deletes the specified port channel, its associated interface mappings, and the hardware associations for this SAN port channel.

Example

The following example demonstrates how to delete a SAN port channel:

```
switch(config)# no interface san-port-channel 1
```

The SAN port channel 1 is deleted and all its members are disabled. Please do the same operation on the switch at the other end of the SAN port channel.

Interfaces in a SAN Port Channel

You can add or remove a physical Fibre Channel interface (or a range of interfaces) to an existing SAN port channel. The compatible parameters on the configuration are mapped to the SAN port channel. Adding an interface to a SAN port channel increases the channel size and bandwidth of the SAN port channel. Removing an interface from a SAN port channel decreases the channel size and bandwidth of the SAN port channel.

Note Virtual Fibre Channel interfaces cannot be added to SAN port channels.

Adding an Interface to a SAN Port Channel

This section explains how to add an interface to a SAN port channel.

Step 1 switch# **configure terminal**

Enters global configuration mode.

Step 2 switch(config)# **interface** *type slot / port / BO port*

Enters configuration mode for the specified interface.

Step 3 switch(config-if)# **channel-group** *channel-number*

Adds the Fibre Channel interface to the specified channel group. If the channel group does not exist, it is created. The port is shut down.

Break out (BO) port option for Fibre Channel (FC) interfaces is required only for the Cisco Nexus N9K-C9336C-FX2-E platform switch.

Example

The following example adds an interface to a SAN port channel:

```
switch(config)# interface fc9/10
switch(config-if)# channel-group 15
```

fc9/10 is added to san-port-channel 15 and is disabled. Please do the same operation on the switch at the other end of the san-port-channel, then do “no shutdown” at both ends to bring them up

FC support in N9K-9336C-FX2-E switch is available only from port 9 to 36.

Forcing an Interface Addition

You can force the port configuration to be overwritten by the SAN port channel. In this case, the interface is added to a SAN port channel.

Note When SAN port channels are created from within an interface, the **force** option cannot be used.

This section explains how to force the addition of a port to a SAN port channel.

Step 1 switch# **configure terminal**

Enters global configuration mode.

Step 2 switch(config)#**interface** *type slot / port / BO port*

Enters configuration mode for the specified interface.

Step 3 switch(config-if)# **channel-group** *channel-number* **force**

Forces the addition of the interface into the specified channel group. The E port is shut down.

Example

The following example adds an interface to a SAN port channel:

```
switch(config)# interface fc9/10
switch(config-if)# channel-group 15 force
```

fc9/10 added to san-port-channel 15 and disabled. Please do the same operation on the switch at the other end of the san-port-channel, then perform a **no shutdown** at both ends to bring them up

FC support in N9K-9336C-FX2-E switch is available only from port 9 to 36.

About Interface Deletion from a SAN Port Channel

When a physical interface is deleted from the SAN port channel, the channel membership is automatically updated. If the deleted interface is the last operational interface, then the port channel status is changed to a down state. Deleting an interface from a SAN port channel decreases the channel size and bandwidth of the SAN port channel.

Deleting an Interface from a SAN Port Channel

This section explains how to delete a physical interface (or a range of physical interfaces) from a SAN port channel.

Step 1 switch(config)# **interface** *type slot /port / BO port*

Enters configuration mode for the specified interface.

Step 2 switch(config)#**shut**

Shuts down the interface before removing the specified channel group.

Step 3 switch(config)#**no channel-group** *channel-number*

Deletes the physical Fibre Channel interface from the specified channel group.

Break out (BO) port option for Fibre Channel (FC) interfaces is required only for the Cisco Nexus N9K-C9336C-FX2-E platform switch.

Example

The following example deletes an interface from a SAN port channel:

```
switch(config)# interface fc9/10
switch(config-if)# shut
switch(config-if)# no channel-group 15
```

fc9/10 is removed from the SAN port-channel 15 and disabled.

Please do the same operation on the switch at the other end of the san-port-channel

FC support in N9K-9336C-FX2-E switch is available only from port 9 to 36.

Verifying SAN Port Channel Configurations

You can view specific information about existing SAN port channels at any time from EXEC mode. The following **show** commands provide further details on existing SAN port channels.

The **show san-port-channel summary** command displays a summary of SAN port channels within the switch. A one-line summary of each SAN port channel provides the administrative state, the operational state, the number of attached and active interfaces (up), and the first operational port (FOP), which is the primary operational interface selected in the SAN port channel to carry control-plane traffic (no load-balancing). The FOP is the first port that comes up in a SAN port channel and can change if the port goes down. The FOP is also identified by an asterisk (*).

To display VSAN configuration information, perform one of the following tasks:

Step 1 switch# **show san-port-channel summary** | **database** | **consistency** [*details*] | **usage** | **compatibility-parameters**

Displays SAN port channel information.

Step 2 switch# **show san-port-channel database interface san-port-channel** *channel-number*

Displays information for the specified SAN port channel.

Step 3 switch# **show interface type slot** */ port / BO port*

Displays VSAN configuration information for the specified Fibre Channel interface.

Break out (BO) port option for Fibre Channel (FC) interfaces is required only for the Cisco Nexus N9K-C9336C-FX2-E platform switch.

Example

The following example shows how to display a summary of SAN port channel information:

```
switch# show san-port-channel summary
--------------------------------------------------------------------------------
Interface    Total Ports Oper Ports First Oper Port-
--------------------------------------------------------------------------------
san-port-channel     7         2         0     -
san-port-channel     8         2         0     -
san-port-channel     9         2         2
```

The following example shows how to display SAN port channel consistency:

```
switch# show san-port-channel consistency
Database is consistent
```

The following example shows how to display details of the used and unused port channel numbers:

```
switch# show san-port-channel usage
Totally 3 port-channel numbers used
```

```
================================
Used : 77 - 79
Unused: 1 - 76, 80 - 256
```

FLOGI Operation

When an NP port becomes operational, the switch first logs itself in to the core switch by sending a FLOGI request (using the port WWN of the NP port).

After completing the FLOGI request, the switch registers itself with the fabric name server on the core switch (using the symbolic port name of the NP port and the IP address of the edge switch).

The following table identifies port and node names in the edge switch used in FC NPV mode.

Table 4: Edge Switch FLOGI Parameters

Parameter	Derived From
pWWN	The fWWN of the NP port on the edge switch.
nWWN	The VSAN-based sWWN of the edge switch.
symbolic port name	The edge switch name and NP port interface string. **Note** If no switch name is available, the output will read "switch." For example, switch: fc 1/5.
IP address	The IP address of the edge switch.
symbolic node name	The edge switch name.

NPV Traffic Management

Automatic Uplink Selection

NPV supports automatic selection of NP uplinks. When a server interface is brought up, the NP uplink interface with the minimum load is selected from the available NP uplinks in the same VSAN as the server interface.

When a new NP uplink interface becomes operational, the existing load is not redistributed automatically to include the newly available uplink. Server interfaces that become operational after the NP uplink can select the new NP uplink.

Traffic Maps

FC NPV supports traffic maps. A traffic map allows you to specify the NP uplinks that a server interface can use to connect to the core switches.

Note When an FC NPV traffic map is configured for a server interface, the server interface must select only from the NP uplinks in its traffic map. If none of the specified NP uplinks are operational, the server remains in a non-operational state.

The FC NPV traffic map feature provides the following benefits:

- Facilitates traffic engineering by allowing configuration of a fixed set of NP uplinks for a specific server interface (or range of server interfaces).
- Ensures correct operation of the persistent FC ID feature; this is because a server interface will always connect to the same NP uplink (or one of a specified set of NP uplinks) after an interface reinitialization or switch reboot.

Disruptive Auto Load Balancing of Server Logins across NP Links

FC NPV supports disruptive load balancing of server logins. When disruptive load balancing is enabled, FC NPV redistributes the server interfaces across all available NP uplinks when a new NP uplink becomes operational. To move a server interface from one NP uplink to another NP uplink, FC NPV forces reinitialization of the server interface so that the server performs a new login to the core switch.

In Release 7.0(3)I7(2) and later software releases, FC NPV supports disruptive load balancing of server logins. When disruptive load balancing is enabled, FC NPV redistributes the server interfaces across all available NP uplinks when a new NP uplink becomes operational. To move a server interface from one NP uplink to another NP uplink, FC NPV forces reinitialization of the server interface so that the server performs a new login to the core switch.

Only server interfaces that are moved to a different uplink are reinitialized. A system message is generated for each server interface that is moved.

Note Redistributing a server interface causes traffic disruption to the attached end devices. Adding a member to the existing port-channel does not trigger disruptive auto load-balance.

To avoid disruption of server traffic, you should enable this feature only after adding a new NP uplink, and then disable it again after the server interfaces have been redistributed.

If disruptive load balancing is not enabled, you can manually reinitialize some or all of the server interfaces to distribute server traffic to new NP uplink interfaces.

FC NPV Traffic Management Guidelines

When deploying FC NPV traffic management, follow these guidelines:

- Use FC NPV traffic management only when automatic traffic engineering does not meet your network requirements.
- You do not need to configure traffic maps for all server interfaces. By default, FC NPV will use automatic traffic management.

- Server interfaces configured to use a set of NP uplink interfaces cannot use any other available NP uplink interfaces, even if none of the configured interfaces are available.
- When disruptive load balancing is enabled, a server interface may be moved from one NP uplink to another NP uplink. Moving between NP uplink interfaces requires FC NPV to relogin to the core switch, causing traffic disruption.
- To link a set of servers to a specific core switch, associate the server interfaces with a set of NP uplink interfaces that all connect to that core switch.
- Configure Persistent FC IDs on the core switch and use the Traffic Map feature to direct server interface traffic onto NP uplinks that all connect to the associated core switch.

FC NPV Guidelines and Limitations

When configuring FC NPV, note the following guidelines and limitations:

- Fibre Channel N-port Virtualization (NPV) can co-exist with VXLAN on different fabric uplinks but on same or different front panel ports on the Cisco Nexus 93180YC-FX, N9K-C9336C-FX2-E, and N9K-C93360YC-FX2 switches. VXLAN can only exist on the Ethernet front panel ports but not on the FC front panel ports.
- In-order data delivery is not required in FC NPV mode because the exchange between two end devices always takes the same uplink from the edge switch to the core. Upstream of the edge switch, core switches will enforce in-order delivery if configured.
- You can configure zoning for end devices that are connected to edge switches using all the available member types on the core switch. However, the preferred way of zoning servers connected to any switch in NPV mode is via PWWN, device-alias and fcalias. You must place multiple servers in the same zone only when using smart zoning. For more information about smart zoning on Cisco MDS switches, refer the chapter *Configuring and Managing Zones* in *Cisco MDS 9000 Series Fabric Configuration Guide* .
- Port tracking is not supported in FC NPV mode.
- Port security is supported on the core switch for devices logged in through the FC NPV switch. Port security is enabled on the core switch on a per-interface basis. To enable port security on the core switch for devices that log in through an FC NPV switch, you must adhere to the following requirements:
  - The internal FLOGI must be in the port security database; in this way, the port on the core switch will allow communications and links.
  - All the end device pWWNs must also be in the port security database.
- Edge switches can connect to multiple core switches. In other words, different NP ports can be connected to different core switches.
- If a server interface goes down and then returns to service, the interface is not guaranteed to be assigned to the same NP uplink.
- The server interface is only operational when its assigned NP uplink is operational.
- Both servers and targets can be connected to the switch when in FC NPV mode.
- Fibre Channel switching is not performed in the edge switch; all traffic is switched in the core switch.
- FC NPV supports NPIV-capable servers. This capability is called nested NPIV.

- Connecting two Cisco FC NPV switches together is not supported.
- Only F and NP ports are supported in FC NPV mode.
- **Speed auto-negotiation** is supported only for Cisco Nexus 93180YC-FXand N9k-C93360YC-FX2 switches. The default speed is set to auto.
- Speed auto-negotiation is not supported in N9K-C9336C-FX2-E switch and default speed is set to 32G.
- Nexus 9000 only supports the IDLE fill pattern on 8 Gbps Fibre Channel interfaces. For Nexus 9000 FC interface to operate at 8 Gbps, peer device must be configured to use a matching IDLE fill pattern. Most server and target FC interfaces do not support this and thus cannot connect to Nexus 9000 at 8 Gbps. To interoperate with other Fibre Channel switches at 8 Gbps ensure the peer switch FC interface also uses a matching IDLE fill pattern. For Cisco MDS switches, configure using the **switchport fill-pattern** interface configuration command. To connect to a peer Nexus 9000 at 8 Gbps, use no fill pattern configuration, as both devices use matching IDLE fill patterns by default.
- The default port-speed for all FC interfaces is auto for Cisco Nexus N9k-C93180YC-FX and N9k-C93360YC-FX2 switches.
- The default port-speed of all FC ports is 32G for Cisco Nexus N9K-C9336C-FX2-E switch.
- The receive B2B credit value is 64 in N9K-C93180YC-FX and 32 in N9K-C93360YC-FX2 and N9K-C9336C-FX2-E. This is not configurable.
- When a san-port channel is created, it is created in **channel mode active** by default; **channel mode on** is not supported for NPV switch.
- vFC flap may be required to bring up N Port vFC interfaces after changing the FCoE FC map.
- FC-NPV (up to 32G) and FCoE-NPV are supported on N9K-C93180YC-FX, N9K-C9336C-FX2-E, and N9k-C93360YC-FX2 switches both as NP uplink and F host port.
- Beginning with Cisco NX-OS Release 10.2(2)F, FC-NPV is supported on Cisco Nexus N9K-C9336C-FX2-E switch.
- Operating speed and member addition to san-po limitation on N9K-C9336C-FX2-E:

  1. Speed change of fc-breakout
     - Default speed is 32G
     - Speed change cannot be done on a single fc-breakout interface level
     - Speed change of fc-breakout is done on range of fc-breakout interface level
       - The range should contain full set of fc-breakout corresponds to a front panel port - for any partial range, speed config throws ERR_01 error.
       - The range should not contain any fc-breakout which is part of san-po - if range has any san-po member, speed config throws ERR_02 error.
       - The range can have fc-breakout ports corresponds to multiple front panel ports.

  2. Speed change of san-po
     - Default speed is 32G.

- Speed change of san-po is allowed only if its members includes all fc-breakout ports corresponds to a front panel port - if san-po has partial set fc-breakout ports corresponds a front panel port, speed change throws ERR_03 error.
- Speed change of san-po can be done by giving range of san-po interfaces.

3. Speed config in running config
   - Speed config(not default one) will be displayed in fc-breakout interface range level; it will not be displayed under individual fc-breakout interface for the `"sh runn"` command. Speed config(not default one) will be displayed in the `"show interface fc<int no>"` command.
4. Member addition to san-po (channel-group x)
   - The interface range should contain full set of fc-breakout corresponds to a front panel port - for any partial range, though the channel addition is successful, warningWARN_01 message will be thrown.
   - The range can have fc-breakout ports corresponds to multiple front panel ports.

- Error and Warning messages:
  - ERR_01:

    ```
    if-range contains partial set of fc1/18/1-4 fc-breakout ports
    ```
  - ERR_02:

    ```
    if-range contains fc1/21/1-4 ports; some are part sanpo
    ```
  - ERR_03:

    ```
    san-port-channel21 does not contain full set of fc1/22/1-4 fc-breakout ports
    ```
  - WARN_01:

    ```
    Warning: if-range contains partial set of fc1/22/1-4 fc-breakout ports
    ```

Licensing Requirements for FC NPV

The following table shows the licensing requirements for FC NPV.

Product	Product ID	License Requirement
Cisco NX-OS	• N93-16Y-SSK9 • N93-48Y-SSK9 • ACI-STRG	FC NPV requires both the following licenses: • SAN_ENTERPRISE_PKG—A feature license to activate FC and FCoE NPV. • FC_PORT_ACTIVATION_PKG—The number of ports to activate for FC. This comes in two variants (16 port and 48 port). **Note** For a complete explanation of the Cisco NX-OS licensing scheme and how to obtain and apply licensees, see the Cisco NX-OS Licensing Guide .

Configuring NPV

Enabling FC NPV

FC NPV is enabled when **feature-set fcoe-npv** is installed and enabled.

To enable **fcoe-npv**, perform this task:

Note This enables both FC and FCoE NPV mode.

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **install feature-set fcoe-npv**
3. switch(config-npv)# **feature-set fcoe-npv**

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters configuration mode.
Step 2	switch(config)# **install feature-set fcoe-npv**	Installs the FC and FCoE NPV feature set.
Step 3	switch(config-npv)# **feature-set fcoe-npv**	Enables FC and FCoE NPV.

Converting Ethernet Ports to Fibre Channel

This section explains how to convert Ethernet ports to fibre channel ports.

Before you begin

This task requires installing and enabling the port license.

Step 1 Perform TCAM carving.

Example:

```
Switch(config)# hardware access-list tcam region ing-racl 1536
Switch(config)# hardware access-list tcam region ing-ifacl 256
Switch(config)# hardware access-list tcam region ing-redirect 256
```

Step 2 Confirm that **feature-set fcoe-npv** is installed and enabled.

Example:

```
Switch(config)# install feature-set fcoe-npv
Switch(config)# feature-set fcoe-npv
```

Step 3 Convert the port(s) to FC.

Example:

In this example, an Ethernet interface is being converted to FC interface on Cisco Nexus 9300-FX switches.

```
Switch(config)# slot 1
Switch(config)# port 1-4,45-48 type fc
Port type is changed. ACTION REQUIRED: Please save configurations and reload the switch
```

Note You must convert all the four front panel ports in a column to FC/Ethernet together.

In this example, an Ethernet interface is being converted to FC interface on Cisco Nexus N9K-93360YC-FX2 switches. In this switch, four ports form a port group. For example the first port group will be 1,2,49,50; the second port group will be 3,4,51,52 and likewise.

```
Switch(config)# slot 1
Switch(config)# port 1-2, 49-50 type fc
Switch(config)# port 3-4, 51-52 type fc
Port type is changed. ACTION REQUIRED: Please save configurations and reload the switch
```

In this example, an Ethernet interface is being converted to FC interface on Cisco Nexus N9K-9336C-FX2-E switches.

```
Switch(config)# slot 1
Switch(config)# port 9,12,33 type fc breakout
Port type is changed. ACTION REQUIRED: Please save configurations and reload the switch
```

Step 4 Convert the FC interface back to Ethernet port(s).

Example:

In this example, an FC interface is being converted back to Ethernet interface on Cisco Nexus 9300-FX switches.

```
Switch(config)# slot 1
Switch(config)# port 1-4,45-48 type eth
Port type is changed. ACTION REQUIRED: Please save configurations and reload the switch
```

In this example, an FC interface is being converted back to Ethernet interface on a Cisco Nexus N9K-93360YC-FX2 switch.

```
Switch(config)# slot 1
Switch(config)# port 1-2, 49-50 type eth
Port type is changed. ACTION REQUIRED: Please save configurations and reload the switch
```

In this example, an FC interface is being converted back to Ethernet interface on Cisco Nexus N9K-9336CFX2-E switches.

Note Port 1-8 cannot be converted to FC in N9K-C9336C-FX2-E.

```
Switch(config)# slot 1
Switch(config)# port 9,12,33 type eth
Port type is changed. ACTION REQUIRED: Please save configurations and reload the switch
```

After the conversion, save the configuration and reload the switch.

Note In Cisco Nexus 93180YC-FX, ports can be converted only in groups (sequential) of 4 (in multiples of 4).

Enabling the Fibre Channel Port License

This section explains how to enable the licensing for FC NPV.

Before you begin

To enable the port license, you must shut down the fibre channel (FC) ports.

Enable the port license.

Example:

```
Switch(config)# int fc1/1
Switch(config-if)# port-license acquire
```

Note This step is required during bring-up of native FC port.

Configuring FC NPV Interfaces

After you enable FC NPV, you should configure the NP uplink interfaces and the server interfaces.

Configuring FC NP Interfaces

To configure an NP uplink interface, perform this task:

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **interface** { **fc** *slot/port/BO port* | **san-port-channel** <*number*> }
3. switch(config-if)# **switchport speed** *speed*
4. switch(config-if)# **switchport mode NP**
5. switch(config-if)# **no shutdown**

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **interface** { **fc** *slot/port/BO port* | **san-port-channel** <*number*> }	Selects an interface (Fibre Channel or SAN port channel) that will be connected to the core FC NPV switch.
Step 3	switch(config-if)# **switchport speed** *speed*	Sets the speed, which can be 4G, 8G, 16G, 32G or auto.

	Command or Action	Purpose

		Note For 8G NP link, the fill-pattern should be set to IDLE on the core switch.
		4G and auto speed is not supported on Cisco N9K-C9336C-FX2-E switches.
		The following is an example of configuring IDLE fill pattern on a Cisco MDS switch:
		`Switch(config)# int fc2/3`
		`Switch(config)# switchport fill-pattern IDLE speed 8000`
		`Switch(config)# sh run int fc2/3`
		`interface fc2/3`
		`switchport speed 8000`
		`switchport mode NP`
		`switchport fill-pattern IDLE speed 8000`
		`no shutdown`

Step 4	switch(config-if)# **switchport mode NP**	Configures the interface as an NP port.
Step 5	switch(config-if)# **no shutdown**	Brings up the interface.

Configuring a Server Interface

To configure a server interface, perform this task:

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **interface fc** *slot*/*port*
3. switch(config-if)# **switchport speed** *speed*
4. switch(config-if)# **switchport mode F**
5. switch(config-if)# **no shutdown**

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **interface fc** *slot*/*port*	Creates an interface that connects the server to the NPV switch.
Step 3	switch(config-if)# **switchport speed** *speed*	Sets the speed, which can be 4G, 8G, 16G, 32G or auto. **Note** 8G speed is not supported for server and target interfaces.
Step 4	switch(config-if)# **switchport mode F**	Configures the interface as an F port.
Step 5	switch(config-if)# **no shutdown**	Brings up the interface.

Configuring NPV Traffic Management

Configuring NPV Traffic Maps

An NPV traffic map associates one or more NP uplink interfaces with a server interface. The switch associates the server interface with one of these NP uplinks.

Note To map the server interface to a different uplink, the server interface must be shut down before configuring the traffic map.

To configure a traffic map, perform this task:

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **npv traffic-map server-interface** {**fc** *slot/port* | **vfc** *vfc-id*} **external-interface** { **fc** *slot/port* | **san-port-channel** <*number*> | **vfc** *vfc-id* | **vfc-port-channel** *vfc-port-channel-id* }
3. switch(config)# **no npv traffic-map server-interface** {**fc** *slot/port* | **vfc** *vfc-id*} **external-interface** { **fc** *slot/port* | **san-port-channel** <*number*> | **vfc** *vfc-id* | **vfc-port-channel** *vfc-port-channel-id* }

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **npv traffic-map server-interface** {**fc** *slot/port* | **vfc** *vfc-id*} **external-interface** { **fc** *slot/port* | **san-port-channel** <*number*> | **vfc** *vfc-id* | **vfc-port-channel** *vfc-port-channel-id* }	Configures a mapping between a server interface (or range of server interfaces) and an NP uplink interface (or range of NP uplink interfaces). **Note** To map the server interface to a different uplink, the server interface must be shut down before configuring the traffic map.
Step 3	switch(config)# **no npv traffic-map server-interface** {**fc** *slot/port* | **vfc** *vfc-id*} **external-interface** { **fc** *slot/port* | **san-port-channel** <*number*> | **vfc** *vfc-id* | **vfc-port-channel** *vfc-port-channel-id* }	Removes the mapping between the specified server interfaces and NP uplink interfaces.

Enabling Disruptive Load Balancing

If you configure additional NP uplinks, you can enable the disruptive load-balancing feature to distribute the server traffic load evenly among all the NP uplinks.

To enable disruptive load balancing, perform this task:

SUMMARY STEPS

1. switch# **configure terminal**
2. switch(config)# **npv auto-load-balance disruptive**

3. switch (config)# **no npv auto-load-balance disruptive**

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters configuration mode on the NPV.
Step 2	switch(config)# **npv auto-load-balance disruptive**	Enables disruptive load balancing on the switch.
Step 3	switch (config)# **no npv auto-load-balance disruptive**	Disables disruptive load balancing on the switch.

Verifying FC NPV

To display information about FC NPV, perform the following task:

SUMMARY STEPS

1. switch# **show feature-set | i fcoe**
2. switch# **show npv flogi-table** [**all**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch# **show feature-set	i fcoe** **Example:** `switch# show feature-set	i fcoe` `fcoe-npv 8 enabled`	
Step 2	switch# **show npv flogi-table** [**all**]	Displays the FC NPV configuration.

Verifying FC NPV Examples

To display a list of devices on a server interface and their assigned NP uplinks, enter the **show npv flogi-table** command on the Cisco Nexus 9000 Series switch:

```
switch# show npv flogi-table
--------------------------------------------------------------------------------
SERVER                                                                 EXTERNAL
INTERFACE VSAN FCID            PORT NAME               NODE NAME       INTERFACE
--------------------------------------------------------------------------------
vfc1/x    1    0xee0008 10:00:00:00:c9:60:e4:9a 20:00:00:00:c9:60:e4:9a  fc1/x
vfc1/x    1    0xee0009 20:00:00:00:0a:00:00:01 20:00:00:00:c9:60:e4:9a  fc1/x
vfc1/x    1    0xee000a 20:00:00:00:0a:00:00:02 20:00:00:00:c9:60:e4:9a  fc1/x
vfc1/x    1    0xee000b 33:33:33:33:33:33:33:33 20:00:00:00:c9:60:e4:9a  fc1/x

Total number of flogi = 4
```

Note For each server interface, the External Interface value displays the assigned NP uplink.

To display the status of the server interfaces and the NP uplink interfaces, enter the **show npv status** command:

```
switch# show npv status

npiv is enabled

disruptive load balancing is disabled

External Interfaces:
====================
  Interface: fc1/47, State: Down
  Interface: san-port-channel 200, State: Trunking
        VSAN:    1, State: Up
        VSAN:  200, State: Up
        VSAN:  201, State: Up
        VSAN:  202, State: Up, FCID: 0xea0020
        VSAN:  100, State: Up
        VSAN:   55, State: Up
  Interface: vfc-po149, State: Trunking
        VSAN:  201, State: Up
        VSAN:  202, State: Up, FCID: 0xea0260
        VSAN:  100, State: Up
  Interface: vfc-po4090, State: Trunking
        VSAN:  201, State: Up
        VSAN:  202, State: Up, FCID: 0xea0220
        VSAN:  100, State: Up
  Interface: vfc1/9, State: Trunking
        VSAN:  201, State: Up
        VSAN:  202, State: Up, FCID: 0xea0240
        VSAN:  100, State: Up

  Number of External Interfaces: 5

Server Interfaces:
==================
  Interface: fc1/38, VSAN:  100, State: Up
  Interface: fc1/39, VSAN:  202, State: Up
  Interface: fc1/40, VSAN: 4094, State: Down
  Interface: vfc100, VSAN: 4094, State: Down
  Interface: vfc151, VSAN: 4094, State: Down
  Interface: vfc1/14, VSAN:  100, State: Up

  Number of Server Interfaces: 6
```

Note To view fcns database entries for FC NPV edge switches, you must enter the **show fcns database** command on the core switch.

To view all the FC NPV edge switches, enter the **show fcns database** command on the core switch:

```
core-switch# show fcns database
```

For additional details (such as IP addresses, switch names, interface names) about the FC NPV edge switches that you see in the **show fcns database** output, enter the **show fcns database detail** command on the core switch:

```
core-switch# show fcns database detail
========================================================================
-----------------------
VSAN:100   FCID:0xe101c0
-----------------------
port-wwn (vendor)           :50:0a:09:82:ad:0d:86:37 (NetApp)
node-wwn                    :50:0a:09:80:8d:0d:86:37
class                       :3
node-ip-addr                :0.0.0.0
ipa                         :00 00 00 00 1e 22 a0 00
fc4-types:fc4_features      :scsi-fcp:target
symbolic-port-name          :NetApp FC Target Adapter (8112) lab-D-netapp01:3b
symbolic-node-name          :NetApp FAS3240 (lab-D-netapp01)
port-type                   :N
port-ip-addr                :0.0.0.0
fabric-port-wwn             :21:61:00:2a:6a:5b:da:00
hard-addr                   :0x000000
permanent-port-wwn (vendor) :50:0a:09:82:ad:0d:86:37 (NetApp)
connected interface         :vfc6/33
switch name (IP address)    :MDS9706 (10.105.188.173)
-----------------------
VSAN:100   FCID:0xe101ef
-----------------------
port-wwn (vendor)           :50:06:01:6b:08:60:7c:71 (Clariion)
node-wwn                    :50:06:01:60:88:60:7c:71
class                       :3
node-ip-addr                :0.0.0.0
ipa                         :ff ff ff ff ff ff ff ff
fc4-types:fc4_features      :scsi-fcp:both
symbolic-port-name          :CLARiiON:::SPB23::FC::::::
symbolic-node-name          :CLARiiON:::SPB::FC::::::
port-type                   :N
port-ip-addr                :0.0.0.0
fabric-port-wwn             :20:19:00:2a:6a:5b:da:00
hard-addr                   :0x000000
permanent-port-wwn (vendor) :50:06:01:6b:08:60:7c:71 (Clariion)
connected interface         :fc1/25
switch name (IP address)    :MDS9706 (10.105.188.173)

core-switch# show interface fc 1/1
fc1/1 is trunking
    Hardware is Fibre Channel, SFP is short wave laser w/o OFC (SN)
    Port WWN is 20:01:2c:d0:2d:50:d2:a0
    Admin port mode is NP, trunk mode is on
    snmp link state traps are enabled
    Port mode is TNP
    Port vsan is 201
    Speed is 16 Gbps
    Transmit B2B Credit is 500
    Receive B2B Credit is 64
    Receive data field Size is 2112
    Beacon is turned off
    Belongs to san-port-channel 200
    Trunk vsans (admin allowed and active) (1,55,100,200-202,204)
    Trunk vsans (up)                       (100,202)
    Trunk vsans (isolated)                 (204)
    Trunk vsans (initializing)             (1,55,200-201)
    5 minutes input rate 0 bits/sec,0 bytes/sec, 0 frames/sec
    5 minutes output rate 0 bits/sec,0 bytes/sec, 0 frames/sec
      406 frames input,40164 bytes
        0 discards,0 errors
        0 invalid CRC/FCS,0 unknown class
        0 too long,0 too short
      192 frames output,14364 bytes
```

```
    0 discards,0 errors
  1 input OLS,1  LRR,5 NOS,0 loop inits
  3 output OLS,1 LRR, 4 NOS, 0 loop inits
  500 transmit B2B credit remaining
  0 low priority transmit B2B credit remaining
Last clearing of "show interface" counters  :never
```

Verifying FC NPV Traffic Management

To display the FC NPV traffic map, enter the **show npv traffic-map** command.

```
switch# show npv traffic-map
NPV Traffic Map Information:
----------------------------------------
Server-If       External-If(s)
----------------------------------------
fc1/3           fc1/10,fc1/11
fc1/5           fc1/1,fc1/2
----------------------------------------
```

To display the FC NPV internal traffic details, enter the **show npv internal info traffic-map** command.

Verifying Disruptive Load Balancing

To display the disruptive load-balancing status, enter the **show npv status** command:

```
switch# show npv status
npiv is enabled
disruptive load balancing is enabled
External Interfaces:
====================
  Interface: fc1/1, VSAN: 2, FCID: 0x1c0000, State: Up
...
```

FC NPV Core Switch and FC NPV Edge Switch Configuration Example

Before you begin

This section demonstrates how to configure FC NPV core and edge switches.

Step 1 Procure and install the SAN_ENTERPRISE_PKG and PORT_ACTIVATION_PKG licenses.

Note The license file is in the .lic format and has to be copied to the switch and installed using the following command:

```
Switch# install license bootflash:Switch_port_lic_48.lic
```

Step 2 Check out the license:

```
Switch(config)# install feature-set fcoe-npv
Switch(config-vdc)# feature-set fcoe-npv
```

Step 3 Configure the needed features on the NPV:

```
Switch(config)# feature telnet
Switch(config)# feature lacp
Switch(config)# feature lldp
```

Step 4 Convert the FC port:

```
Switch(config)# slot 1
Switch(config-slot)# port 13-36 type fc
Port type is changed. ACTION REQUIRED: Please save configurations and reload the switch
```

Step 5 Configure service policies:

```
Switch(config)# system qos
Switch(config-sys-qos)# service-policy type network-qos default-fcoe-8q-nq-policy
Switch(config-sys-qos)# service-policy type queuing output default-fcoe-8q-out-policy
```

Step 6 Configure TCAM carving:

```
Switch(config-vrf)# hardware access-list tcam region ing-racl 1536
Warning: Please save config and reload the system for the configuration to take effect
Switch(config)# hardware access-list tcam region ing-redirect 256
Warning: Please save config and reload the system for the configuration to take effect
```

Step 7 Copy the running configuration to startup:

```
Switch(config)# copy running-config startup-config
[########################################] 100%
```

Step 8 (Mandatory) Reload the switch so that the port conversion is applied and TCAMS are carved properly:

```
Switch(config)# reload
This command will reboot the system. (y/n)?  [n] y
2017 Sep 14 10:12:19 Switch %PLATFORM-2-PFM_SYSTEM_RESET: Manual system restart from Command Line
Interface
```

Step 9 Configure VLAN-VSAN mappings:

```
Switch(config)# vlan 1,20,30,40,1000,1002,1010
Switch(config-vlan)# vlan 20
Switch(config-vlan)# fcoe vsan 200
Switch(config-vlan)# vlan 30
Switch(config-vlan)# fcoe vsan 300
Switch(config-vlan)# vlan 40
Switch(config-vlan)# fcoe vsan 300
Switch(config)# vsan database
Switch(config-vsan-db)# vsan 40
Switch(config-vsan-db)# vsan 200
Switch(config-vsan-db)# vsan 300
```

Step 10 Configure the port license for FC ports:

```
Switch(config)# interface fc1/6
Switch(config-if)# port-license acquire
```

Note Checks out the port license for FC ports

Step 11 Configure the FC NP interface-facing core (this same configuration must be applied on the core switch with **switchport mode F** or **auto** for the FC interface):

```
Switch(config-if)# interface fc1/6
Switch(config-if)# switchport mode NP
Switch(config-if)# no shutdown
```

Step 12 Configure the virtual FC NP interface-facing core (this same configuration must be applied on the core switch with **switchport mode F** or **auto** for the virtual FC interface):

a) Configure the physical Ethernet interface:

```
Switch(config-if)# interface Ethernet1/7
Switch(config-if)# switchport
Switch(config-if)# switchport mode trunk
Switch(config-if)# service-policy type qos input default-fcoe-in-policy
Switch(config-if)# mtu 9216
Switch(config-if)# no shutdown
```

Note The steps *MTU and service-policy* are required only when a Cisco Nexus N9K-C93180YC-F, N9K-C9336C-FX2-E, or N9K-C93360YC-FX2 switch is used as the core switch.

b) Configure the virtual FC interface:

```
Switch(config-if)# interface vfc17
Switch(config-if)# bind interface ethernet1/7
Switch(config-if)# switchport mode NP
Switch(config-if)# no shutdown
```

Step 13 Configure the SAN port channel interface-facing core (This same configuration must be applied on the core switch with **switchport mode F** or **auto** for the port-channel interface. The SAN port-channel number can be different.):

a) Configure the SAN port channel:

```
Switch(config)# interface san-port-channel 250
Switch(config-if)# channel mode active
Switch(config-if)# switchport mode NP
Switch(config-if)# switchport trunk mode on
```

b) Add a member to the SAN port channel:

```
Switch(config-if)# interface fc1/13
Switch(config-if)# port-license acquire (this checks out the port license for FC ports)
Switch(config-if)# switchport trunk mode on
Switch(config-if)# channel-group 250 force
fc1/13 added to port-channel 250 and disabled
Please do the same operation on the switch at the other end of the port-channel,
then do "no shutdown" at both ends to bring it up
Switch(config-if)# no shutdown
```

Step 14 Configure the vFC port channel interface-facing core (This same configuration must be applied on the core switch with **switchport mode F** or **auto** for the virtual FC port-channel interface. The vFC port-channel number can be different):

a) Configure the Ethernet port-channel interface:

```
Switch(config)# interface port-channel500
Switch(config-if)# switchport
Switch(config-if)# switchport mode trunk
Switch(config-if)# mtu 9216
Switch(config-if)# service-policy type qos input default-fcoe-in-policy
```

Note The steps *MTU and service-policy* are required only when a Cisco Nexus N9K-C93180YC-FX, N9K-C9336C-FX2-E, or N9K-C93360YC-FX2 switch is used as the core switch.

b) Add a member to the Ethernet port channel:

```
Switch(config-if)# interface Ethernet1/4
Switch(config-if)# channel-group 500 mode active
Switch(config-if)# no shutdown
```

c) Create a virtual FC port-channel interface:

```
Switch(config)# interface vfc-po500 (this creates a vFC)
Switch(config-if)# bind interface port-channel500
Switch(config-if)# switchport mode NP
Switch(config-if)# switchport trunk mode on
```

Step 15 Configure the FCoE server interface-facing server:

a) Configure the physical Ethernet interfaces:

```
Switch(config-if)# interface Ethernet1/6
Switch(config-if)# switchport
Switch(config-if)# switchport mode trunk
Switch(config-if)# service-policy type qos input default-fcoe-in-policy
Switch(config-if)# mtu 9216
Switch(config-if)# no shutdown
```

b) Configure a virtual FC interface:

```
Switch(config-if)# interface vfc6
Switch(config-if)# bind interface ethernet1/6
Switch(config-if)# switchport trunk mode on
Switch(config-if)# no shutdown
```

c) Assigning a port VSAN for the virtual FC interface:

```
Switch(config-if)# vsan database (this assigns the port vsan)(config-vsan-db)
Switch(config-vsan-db)# vsan 40 interface vfc6
```

Step 16 Configuring FC server interface

a) Configure FC interface in F mode:

```
Switch(config)# interface fc1/39
Switch(config-if)# switchport mode F
```

b) Apply port vsan for the FC interface:

```
Switch(config)# vsan database
Switch(config-if)# vsan 100 interface fc1/39
```

INDEX