

Storage Feature Guide for QFX10000 Switches

Release
15.1X53



Modified: 2016-05-17

Juniper Networks, Inc.
1133 Innovation Way
Sunnyvale, California 94089
USA
408-745-2000
www.juniper.net

Juniper Networks, Junos, Steel-Belted Radius, NetScreen, and ScreenOS are registered trademarks of Juniper Networks, Inc. in the United States and other countries. The Juniper Networks Logo, the Junos logo, and JunosE are trademarks of Juniper Networks, Inc. All other trademarks, service marks, registered trademarks, or registered service marks are the property of their respective owners.

Juniper Networks assumes no responsibility for any inaccuracies in this document. Juniper Networks reserves the right to change, modify, transfer, or otherwise revise this publication without notice.

Storage Feature Guide for QFX10000 Switches
15.1X53
Copyright © 2016, Juniper Networks, Inc.
All rights reserved.

The information in this document is current as of the date on the title page.

YEAR 2000 NOTICE

Juniper Networks hardware and software products are Year 2000 compliant. Junos OS has no known time-related limitations through the year 2038. However, the NTP application is known to have some difficulty in the year 2036.

END USER LICENSE AGREEMENT

The Juniper Networks product that is the subject of this technical documentation consists of (or is intended for use with) Juniper Networks software. Use of such software is subject to the terms and conditions of the End User License Agreement ("EULA") posted at <http://www.juniper.net/support/eula.html>. By downloading, installing or using such software, you agree to the terms and conditions of that EULA.

Table of Contents

	About the Documentation	xi
	Documentation and Release Notes	xi
	Supported Platforms	xi
	Using the Examples in This Manual	xi
	Merging a Full Example	xii
	Merging a Snippet	xii
	Documentation Conventions	xiii
	Documentation Feedback	xv
	Requesting Technical Support	xv
	Self-Help Online Tools and Resources	xv
	Opening a Case with JTAC	xvi
Chapter 1	Storage Overview	17
	Overview of FIP	17
Part 1	Configuring FCoE on a Transit Switch	
Chapter 2	Using FCoE on a Transit Switch	21
	Understanding FCoE Transit Switch Functionality	22
	Understanding FCoE	26
	FCoE Devices	27
	FCoE Frames	28
	Virtual Links	29
	FCoE VLANs	29
	Configuring VLANs for FCoE Traffic on an FCoE Transit Switch	32
	Troubleshooting Dropped FCoE Traffic	37
	Understanding MC-LAGs on an FCoE Transit Switch	41
	Supported Topology	41
	Transit Switches (Server Access)	43
	MC-LAG Switches (FCoE Aggregation)	43
	FIP Snooping and FCoE Trusted Ports	43
	CoS and Data Center Bridging (DCB)	44
	Example: Configuring CoS for FCoE Transit Switch Traffic Across an MC-LAG	44
	Understanding FCoE and FIP Session High Availability	69
	Troubleshooting Dropped FIP Traffic	70
	Understanding OxID Hash Control for FCoE Traffic Load Balancing on Standalone Switches	72
	Enabling and Disabling CoS OxID Hash Control on Standalone Switches	73

Part 2

Chapter 3

Configuring DCBX and PFC

Using DCBX and PFC	77
Understanding DCB Features and Requirements	78
Lossless Transport	78
PFC	79
Buffer Management	79
Physical Interfaces	79
ETS	79
DCBX	80
Understanding CoS Flow Control (Ethernet PAUSE and PFC)	81
General Information about Ethernet PAUSE and PFC and When to Use Them	81
Ethernet PAUSE	82
Symmetric Flow Control	83
Asymmetric Flow Control	84
PFC	87
Lossless Transport Support Summary	90
Example: Configuring CoS PFC for FCoE Traffic	92
Understanding DCBX	101
DCBX Basics	101
DCBX Modes and Support	103
DCBX Modes (Versions)	103
Autonegotiation	105
CNA Support for DCBX Modes	105
Interface Support for DCBX	105
DCBX Attribute Types	105
Asymmetric Attributes	106
Symmetric Attributes	106
DCBX Application Protocol TLV Exchange	107
Application Protocol TLV Exchange	107
FCoE Application Protocol TLV Exchange	107
Disabling Application Protocol TLV Exchange	108
DCBX and PFC	108
DCBX and ETS	108
Default DCBX ETS Advertisement	108
ETS Advertisement and Peer Configuration	109
ETS Recommendation TLV	109
Configuring the DCBX Mode	111
Configuring DCBX Autonegotiation	112
Disabling the ETS Recommendation TLV	115
Understanding DCBX Application Protocol TLV Exchange	116
Applications	116
Application Maps	117
Classifying and Prioritizing Application Traffic	118
Enabling Interfaces to Exchange Application Protocol Information	119
Disabling DCBX Application Protocol Exchange	119
Defining an Application for DCBX Application Protocol TLV Exchange	120

	Configuring an Application Map for DCBX Application Protocol TLV Exchange	121
	Applying an Application Map to an Interface for DCBX Application Protocol TLV Exchange	122
	Example: Configuring DCBX Application Protocol TLV Exchange	123
Chapter 4	Learn About Technology	135
	Data Center Technology Overview Videos	135
	Learn About Video: Why Do We Need an IP Fabric?	135
	Learn About Video: What is the Best Control Plane Protocol to Use in a Data Center IP Fabric?	135
	Learn About Video: Why Use an Overlay Network in a Data Center?	135
	Conceptual Documents That Contain Technology Overview Videos	136
Part 3	Configuration Statements and Operational Commands	
Chapter 5	Configuration Statements (FCoE on a Transit Switch)	139
	family fcoe	140
	oxid	141
Chapter 6	Configuration Statements (DCBX and PFC)	143
	application (Application Maps)	144
	application (Applications)	145
	application-map	146
	application-maps	147
	applications (Applications)	148
	applications (DCBX)	149
	code-points (Application Maps)	150
	dcbx	151
	dcbx-version	152
	destination-port (Applications)	153
	disable (DCBX)	154
	enhanced-transmission-selection	155
	ether-type	156
	interface (DCBX)	157
	no-recommendation-tlv	158
	policy-options	159
	priority-flow-control	160
	protocol (Applications)	161
	recommendation-tlv	162
Chapter 7	Operational Commands (FCoE on a Transit Switch)	163
	clear fip vlan-discovery statistics	164
	show fip vlan-discovery	165
Chapter 8	Operational Commands (DCBX and PFC)	167
	show dcbx	168
	show dcbx neighbors	169

List of Figures

Part 1	Configuring FCoE on a Transit Switch	
Chapter 2	Using FCoE on a Transit Switch	21
	Figure 1: FCoE Transit Switch Connecting FCoE Devices to an FC Switch	24
	Figure 2: ENode Components	28
	Figure 3: Supported Topology for an MC-LAG on an FCoE Transit Switch	42
	Figure 4: Supported Topology for an MC-LAG on an FCoE Transit Switch	47
Part 2	Configuring DCBX and PFC	
Chapter 3	Using DCBX and PFC	77
	Figure 5: PFC for FCoE Traffic Configuration Components Block Diagram	94

List of Tables

	About the Documentation	xi
	Table 1: Notice Icons	xiii
	Table 2: Text and Syntax Conventions	xiii
Part 1	Configuring FCoE on a Transit Switch	
Chapter 2	Using FCoE on a Transit Switch	21
	Table 3: Components of the CoS for FCoE Traffic Across an MC-LAG Configuration Topology	47
Part 2	Configuring DCBX and PFC	
Chapter 3	Using DCBX and PFC	77
	Table 4: Asymmetric Ethernet PAUSE Flow Control Configuration	84
	Table 5: Flow Control State Advertised to the Connected Peer (Autonegotiation)	85
	Table 6: Asymmetric Ethernet PAUSE Behavior on Local and Peer Interfaces	86
	Table 7: Default PFC Priority to Queue and Forwarding Class Mapping	88
	Table 8: Components of the PFC for FCoE Traffic Configuration Topology	93
	Table 9: Summary of Differences Between IEEE DCBX and DCBX Version 1.01	103
	Table 10: Default IEEE 802.1 Classifiers for Trunk Ports and Tagged-Access Ports (Default Trusted Classifier)	124
	Table 11: Default IEEE 802.1 Unicast Classifiers for Access Ports (Default Untrusted Classifier)	125
	Table 12: Components of DCBX Application Protocol Exchange Configuration Topology	126
Part 3	Configuration Statements and Operational Commands	
Chapter 7	Operational Commands (FCoE on a Transit Switch)	163
	Table 13: show fip vlan-discovery Output Fields	165
Chapter 8	Operational Commands (DCBX and PFC)	167
	Table 14: show dcbx output fields	168
	Table 15: show dcbx neighbors Output Fields	169

About the Documentation

- Documentation and Release Notes on page xi
- Supported Platforms on page xi
- Using the Examples in This Manual on page xi
- Documentation Conventions on page xiii
- Documentation Feedback on page xv
- Requesting Technical Support on page xv

Documentation and Release Notes

To obtain the most current version of all Juniper Networks® technical documentation, see the product documentation page on the Juniper Networks website at <http://www.juniper.net/techpubs/>.

If the information in the latest release notes differs from the information in the documentation, follow the product Release Notes.

Juniper Networks Books publishes books by Juniper Networks engineers and subject matter experts. These books go beyond the technical documentation to explore the nuances of network architecture, deployment, and administration. The current list can be viewed at <http://www.juniper.net/books>.

Supported Platforms

For the features described in this document, the following platforms are supported:

- [QFX Series](#)

Using the Examples in This Manual

If you want to use the examples in this manual, you can use the **load merge** or the **load merge relative** command. These commands cause the software to merge the incoming configuration into the current candidate configuration. The example does not become active until you commit the candidate configuration.

If the example configuration contains the top level of the hierarchy (or multiple hierarchies), the example is a *full example*. In this case, use the **load merge** command.

If the example configuration does not start at the top level of the hierarchy, the example is a *snippet*. In this case, use the **load merge relative** command. These procedures are described in the following sections.

Merging a Full Example

To merge a full example, follow these steps:

1. From the HTML or PDF version of the manual, copy a configuration example into a text file, save the file with a name, and copy the file to a directory on your routing platform.

For example, copy the following configuration to a file and name the file **ex-script.conf**. Copy the **ex-script.conf** file to the **/var/tmp** directory on your routing platform.

```
system {
  scripts {
    commit {
      file ex-script.xml;
    }
  }
}
interfaces {
  fxp0 {
    disable;
    unit 0 {
      family inet {
        address 10.0.0.1/24;
      }
    }
  }
}
```

2. Merge the contents of the file into your routing platform configuration by issuing the **load merge** configuration mode command:

```
[edit]
user@host# load merge /var/tmp/ex-script.conf
load complete
```

Merging a Snippet

To merge a snippet, follow these steps:

1. From the HTML or PDF version of the manual, copy a configuration snippet into a text file, save the file with a name, and copy the file to a directory on your routing platform.

For example, copy the following snippet to a file and name the file **ex-script-snippet.conf**. Copy the **ex-script-snippet.conf** file to the **/var/tmp** directory on your routing platform.

```
commit {
  file ex-script-snippet.xml; }
```

2. Move to the hierarchy level that is relevant for this snippet by issuing the following configuration mode command:

```
[edit]
user@host# edit system scripts
[edit system scripts]
```

3. Merge the contents of the file into your routing platform configuration by issuing the **load merge relative** configuration mode command:

```
[edit system scripts]
user@host# load merge relative /var/tmp/ex-script-snippet.conf
load complete
```

For more information about the **load** command, see the *CLI User Guide*.

Documentation Conventions

Table 1 on page xiii defines notice icons used in this guide.

Table 1: Notice Icons

Icon	Meaning	Description
	Informational note	Indicates important features or instructions.
	Caution	Indicates a situation that might result in loss of data or hardware damage.
	Warning	Alerts you to the risk of personal injury or death.
	Laser warning	Alerts you to the risk of personal injury from a laser.
	Tip	Indicates helpful information.
	Best practice	Alerts you to a recommended use or implementation.

Table 2 on page xiii defines the text and syntax conventions used in this guide.

Table 2: Text and Syntax Conventions

Convention	Description	Examples
Bold text like this	Represents text that you type.	To enter configuration mode, type the configure command: user@host> configure

Table 2: Text and Syntax Conventions (*continued*)

Convention	Description	Examples
Fixed-width text like this	Represents output that appears on the terminal screen.	<code>user@host> show chassis alarms</code> <code>No alarms currently active</code>
<i>Italic text like this</i>	<ul style="list-style-type: none">Introduces or emphasizes important new terms.Identifies guide names.Identifies RFC and Internet draft titles.	<ul style="list-style-type: none">A policy <i>term</i> is a named structure that defines match conditions and actions.<i>Junos OS CLI User Guide</i>RFC 1997, <i>BGP Communities Attribute</i>
<i>Italic text like this</i>	Represents variables (options for which you substitute a value) in commands or configuration statements.	Configure the machine's domain name: [edit] root@# set system domain-name <i>domain-name</i>
Text like this	Represents names of configuration statements, commands, files, and directories; configuration hierarchy levels; or labels on routing platform components.	<ul style="list-style-type: none">To configure a stub area, include the stub statement at the [edit protocols ospf area area-id] hierarchy level.The console port is labeled CONSOLE.
< > (angle brackets)	Encloses optional keywords or variables.	stub <default-metric metric>;
(pipe symbol)	Indicates a choice between the mutually exclusive keywords or variables on either side of the symbol. The set of choices is often enclosed in parentheses for clarity.	broadcast multicast (string1 string2 string3)
# (pound sign)	Indicates a comment specified on the same line as the configuration statement to which it applies.	rsvp { # Required for dynamic MPLS only
[] (square brackets)	Encloses a variable for which you can substitute one or more values.	community name members [<i>community-ids</i>]
Indentation and braces ({ })	Identifies a level in the configuration hierarchy.	[edit] routing-options { static { route default { nexthop address; retain; } } }
;(semicolon)	Identifies a leaf statement at a configuration hierarchy level.	
GUI Conventions		
Bold text like this	Represents graphical user interface (GUI) items you click or select.	<ul style="list-style-type: none">In the Logical Interfaces box, select All Interfaces.To cancel the configuration, click Cancel.

Table 2: Text and Syntax Conventions (*continued*)

Convention	Description	Examples
> (bold right angle bracket)	Separates levels in a hierarchy of menu selections.	In the configuration editor hierarchy, select Protocols>Ospf .

Documentation Feedback

We encourage you to provide feedback, comments, and suggestions so that we can improve the documentation. You can provide feedback by using either of the following methods:

- Online feedback rating system—On any page of the Juniper Networks TechLibrary site at <http://www.juniper.net/techpubs/index.html>, simply click the stars to rate the content, and use the pop-up form to provide us with information about your experience. Alternately, you can use the online feedback form at <http://www.juniper.net/techpubs/feedback/>.
- E-mail—Send your comments to techpubs-comments@juniper.net. Include the document or topic name, URL or page number, and software version (if applicable).

Requesting Technical Support

Technical product support is available through the Juniper Networks Technical Assistance Center (JTAC). If you are a customer with an active J-Care or Partner Support Service support contract, or are covered under warranty, and need post-sales technical support, you can access our tools and resources online or open a case with JTAC.

- JTAC policies—For a complete understanding of our JTAC procedures and policies, review the *JTAC User Guide* located at <http://www.juniper.net/us/en/local/pdf/resource-guides/7100059-en.pdf>.
- Product warranties—For product warranty information, visit <http://www.juniper.net/support/warranty/>.
- JTAC hours of operation—The JTAC centers have resources available 24 hours a day, 7 days a week, 365 days a year.

Self-Help Online Tools and Resources

For quick and easy problem resolution, Juniper Networks has designed an online self-service portal called the Customer Support Center (CSC) that provides you with the following features:

- Find CSC offerings: <http://www.juniper.net/customers/support/>
- Search for known bugs: <http://www2.juniper.net/kb/>
- Find product documentation: <http://www.juniper.net/techpubs/>
- Find solutions and answer questions using our Knowledge Base: <http://kb.juniper.net/>

- Download the latest versions of software and review release notes:
<http://www.juniper.net/customers/csc/software/>
- Search technical bulletins for relevant hardware and software notifications:
<http://kb.juniper.net/InfoCenter/>
- Join and participate in the Juniper Networks Community Forum:
<http://www.juniper.net/company/communities/>
- Open a case online in the CSC Case Management tool: <http://www.juniper.net/cm/>

To verify service entitlement by product serial number, use our Serial Number Entitlement (SNE) Tool: <https://tools.juniper.net/SerialNumberEntitlementSearch/>

Opening a Case with JTAC

You can open a case with JTAC on the Web or by telephone.

- Use the Case Management tool in the CSC at <http://www.juniper.net/cm/>.
- Call 1-888-314-JTAC (1-888-314-5822 toll-free in the USA, Canada, and Mexico).

For international or direct-dial options in countries without toll-free numbers, see <http://www.juniper.net/support/requesting-support.html>.

CHAPTER 1

Storage Overview

- Overview of FIP on page 17

Overview of FIP

Fibre Channel over Ethernet (FCoE) Initialization Protocol (FIP) is a Layer 2 protocol that establishes and maintains Fibre Channel (FC) virtual links between pairs of FCoE devices such as server FCoE Nodes (ENodes) and FC switches. FIP can also establish and maintain virtual links between FCoE devices and an FCoE-FC gateway (such as the QFX3500 switch), where the gateway acts on behalf of the FC switch.

FIP enables FCoE devices to discover one another and to initialize and maintain virtual links over a physical Ethernet network. This allows FCoE devices in the Ethernet network to access storage devices in the FC storage area network (SAN).

FIP solves the problem presented by the FC requirement for point-to-point connections (FC does not permit point-to-multipoint connections) by creating a unique virtual link for each connection between an ENode VN_Port and an FC switch VF_Port. Multiple virtual links can use a single physical link and virtual links can traverse Ethernet transit (passthrough) switches while appearing to be direct point-to-point connections to the FC switch.

FIP has its own EtherType (0x8914) to distinguish its traffic from payload-carrying FCoE traffic and other Ethernet traffic. FIP operations occur on a per-VLAN basis.

For more details about FIP, see the Technical Committee T11 organization document *Fibre Channel Backbone - 5 (FC-BB-5) Rev 2.00* available at <http://www.t11.org/ftp/t11/pub/fc/bb-5/09-056v5.pdf>.

PART 1

Configuring FCoE on a Transit Switch

- [Using FCoE on a Transit Switch on page 21](#)

CHAPTER 2

Using FCoE on a Transit Switch

- [Understanding FCoE Transit Switch Functionality on page 22](#)
- [Understanding FCoE on page 26](#)
- [Configuring VLANs for FCoE Traffic on an FCoE Transit Switch on page 32](#)
- [Troubleshooting Dropped FCoE Traffic on page 37](#)
- [Understanding MC-LAGs on an FCoE Transit Switch on page 41](#)
- [Example: Configuring CoS for FCoE Transit Switch Traffic Across an MC-LAG on page 44](#)
- [Understanding FCoE and FIP Session High Availability on page 69](#)
- [Troubleshooting Dropped FIP Traffic on page 70](#)
- [Understanding OxID Hash Control for FCoE Traffic Load Balancing on Standalone Switches on page 72](#)
- [Enabling and Disabling CoS OxID Hash Control on Standalone Switches on page 73](#)

Understanding FCoE Transit Switch Functionality

You can use a QFX Series as a Fibre Channel over Ethernet (FCoE) transit switch. An FCoE transit switch is a Layer 2 data center bridging (DCB) switch that can transport FCoE frames and implements FCoE Initialization Protocol (FIP) snooping. A DCB switch transports both FCoE and Ethernet LAN traffic over the same network infrastructure while preserving the class of service (CoS) that Fibre Channel (FC) traffic requires.

An FCoE transit switch does not encapsulate or de-encapsulate FC frames in Ethernet. It is an access switch that transports FC frames that have already been encapsulated in Ethernet between FCoE initiators such as servers and a storage area network (SAN) FC switch that supports both Ethernet and native FC traffic on its interfaces. The transit switch acts as a passthrough switch and is transparent to the FC switch, which detects each connection to an FCoE device as a direct point-to-point link.

When the QFX Series acts as a transit switch, the VLANs you configure for FCoE traffic can use any of the switch ports on the device (QFX3500 or QFabric system Node device) because the traffic in both directions is standard Ethernet traffic, not native FC traffic.



NOTE: The Ethernet interfaces that connect to FCoE devices must include a native VLAN to transport FIP traffic, because FIP VLAN discovery and notification frames are exchanged as untagged packets. It is a good practice to keep the native VLAN separate from the VLANs that carry FCoE traffic. FCoE VLANs should carry only FCoE traffic, but other types of untagged traffic might use the native VLAN.

Switches and QFabric system Node devices that use the original CLI (not the Enhanced Layer 2 (ELS) software) only require that you configure the native VLAN on the FCoE interfaces that belong to the FCoE VLAN by including the `[set interfaces interface-name unit unit family ethernet-switching native-vlan-id native-vlan-id]` statement in the configuration.

Switches that use ELS software require that you include two statements in the configuration to configure a native VLAN on FCoE interfaces. Include the `[set interfaces interface-name native-vlan-id vlan-id]` statement in the configuration to configure the native VLAN on the interface, and also include the `[set interfaces interface-name unit unit family ethernet-switching native-vlan-id vlan-id]` statement in the configuration to configure the port as a member of the native VLAN.

FCoE traffic should use a VLAN dedicated only to FCoE traffic. You should not mix FCoE traffic with standard Ethernet traffic on a VLAN on the switch.



NOTE: FCoE VLANs (any VLAN that carries FCoE traffic) support only Spanning Tree Protocol (STP) and link aggregation group (LAG) Layer 2 features. IGMP snooping is enabled by default on all VLANs; be sure to disable IGMP snooping on FCoE VLANs.

The transit switch setup differs from the architecture when you configure the switch as an FCoE-FC gateway. (As an FCoE-FC gateway, the switch transports traffic to the FC SAN as native FC frames, and the VLAN must use an FCoE VLAN interface and native FC interfaces to transport that traffic.)

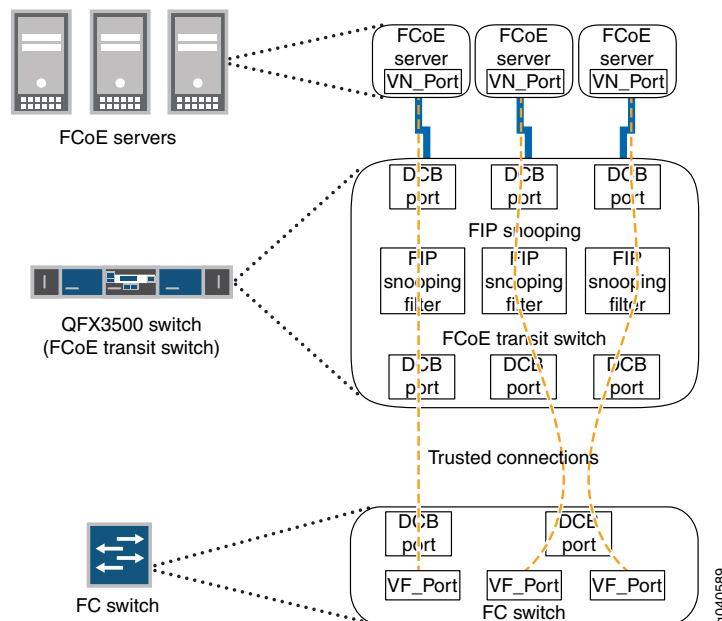


NOTE: On a QFX3500 or QFabric system Node device, the same VLAN cannot be used in both transit switch mode and FCoE-FC gateway mode.

The switch complies with DCB standards for ensuring lossless transport and low latency, and provides 10-Gbps ports for FCoE traffic. For lossless transport to function correctly, you must use priority-based flow control (PFC, described in IEEE 802.1Qbb) to create bandwidth reservations and ensure proper CoS for FCoE traffic. FIP snooping adds security by filtering access so that only traffic from servers that have successfully logged in to the FC network passes through the transit switch and reaches the FC network. To accommodate the larger size of Ethernet-encapsulated frames, FCoE interfaces should be configured with a maximum transmission unit (MTU) size of at least 2180 bytes.

The transit switch transparently connects FCoE-capable devices such as servers in an Ethernet LAN to an FC switch or to a gateway switch (hereafter referred to as the FC switch), as shown in [Figure 1 on page 24](#). The transit switch acts as a transparent DCB access layer between FCoE servers and the FC switch.

Figure 1: FCoE Transit Switch Connecting FCoE Devices to an FC Switch



The transit switch performs FIP snooping at the ports connected to the FCoE devices. At the SAN edge, the FC switch must be able to convert the FCoE traffic to native FC traffic.

Encapsulated FCoE traffic flows through the transit switch to the FCoE ports on the FC switch. The FC switch removes the Ethernet encapsulation from the FCoE frames to

restore the native FC frames. Native FC traffic travels out native FC ports to storage devices in the FC SAN.

Native FC traffic from storage devices flows to the FC switch FC ports, and the FC switch encapsulates that traffic in Ethernet as FCoE traffic. The FCoE traffic flows through the transit switch to the appropriate FCoE device.



NOTE: The FC switch and FC fabric apply appropriate zoning checks on traffic to and from each ENode and provide FC services (for example, name server, fabric login server, or event server).



NOTE: VN_Port to VN_Port FIP snooping is supported to allow FCoE initiators and targets to communicate directly through the switch without going through an FCoE forwarder (FCF) or an FC switch. An FCoE VLAN can support either VN2VF_Port FIP snooping (FC-BB-5) or VN2VN_Port FIP snooping (FC-BB-6), but not both. The same switch can have multiple FCoE VLANs configured, some FCoE VLANs for VN2VF FIP snooping traffic and others for VN2VN FIP snooping traffic.

For load balancing, increasing available bandwidth, and port failover protection, you can configure the 10-Gigabit Ethernet interfaces that belong to an FCoE VLAN as a link aggregation group (LAG). In addition, creating a LAG prevents spanning tree algorithms from blocking physical links and wasting bandwidth.

Related Documentation

- *Overview of Fibre Channel*
- [Understanding DCB Features and Requirements on page 78](#)
- *Understanding an FCoE-FC Gateway*
- [Understanding FCoE on page 26](#)
- *Understanding FCoE LAGs*
- *Understanding VN_Port to VF_Port FIP Snooping on an FCoE Transit Switch*
- *Understanding VN_Port to VN_Port FIP Snooping on an FCoE Transit Switch*
- *Understanding Fibre Channel Terminology*
- [Configuring VLANs for FCoE Traffic on an FCoE Transit Switch on page 32](#)
- *Disabling Enhanced FIP Snooping Scaling*
- *Configuring an FCoE LAG*

Understanding FCoE

Fibre Channel over Ethernet (FCoE) is a method of supporting converged Fibre Channel (FC) and Ethernet traffic on a data center bridging (DCB) network. FCoE encapsulates unmodified FC frames in Ethernet to transport the FC frames over a physical Ethernet network. The T11 Technical Committee, which is the International Committee for Information Technology Standards (INCITS) committee responsible for FC interfaces, developed the FCoE standard to provide a method for transporting FC frames over a DCB network. The T11 document *Fibre Channel Backbone - 5 (FC-BB-5) Rev 2.00* at <http://www.t11.org/ftp/t11/pub/fc/bb-5/09-056v5.pdf> provides details about the FCoE version 1 standard.



NOTE: The switch does not support T11 Annex F *FCoE Pre-FIP Virtual Link Instantiation Protocol*.

To the Ethernet network, an FCoE frame is the same as any other Ethernet frame because the Ethernet encapsulation provides the header information needed to forward the frames. However, to achieve the lossless behavior that FC transport requires, the Ethernet network must conform to DCB standards.

DCB standards create an environment over which FCoE can transport native FC traffic encapsulated in Ethernet while preserving the mandatory class of service (CoS) and other characteristics that FC traffic requires.

Supporting FCoE in a DCB network requires that the FCoE devices in the Ethernet network and the FC switches at the edge of the SAN network handle both Ethernet and native FC traffic. To handle Ethernet traffic, an FC switch does one of two things:

- Incorporates FCoE interfaces.
- Uses an FCoE-FC gateway such as a QFX3500 switch to de-encapsulate FCoE traffic from FCoE devices into native FC and to encapsulate native FC traffic from the FC switch into FCoE and forward it to FCoE devices through the Ethernet network.



NOTE: Standalone switches support FCoE. Virtual Chassis (VC) and mixed-mode Virtual Chassis Fabric (VCF) configurations do not support FCoE. Pure QFX5100 switch VCFs (consisting of only QFX5100 switches) support FCoE.

FCoE concepts include:

- [FCoE Devices on page 27](#)
- [FCoE Frames on page 28](#)
- [Virtual Links on page 29](#)
- [FCoE VLANs on page 29](#)

FCoE Devices

Each FCoE device has a converged network adapter (CNA) that combines the functions of an FC host bus adapter (HBA) and a lossless Ethernet network interface card (NIC) with 10-Gbps Ethernet ports. The portion of the CNA that handles FCoE traffic is called an FCoE Node (ENode). An ENode combines FCoE termination functions and the client part of the FC stack on the CNA.

ENodes present virtual FC interfaces to FC switches in the form of virtual N_Ports (VN_Ports). A VN_Port is an endpoint in a virtual point-to-point connection called a virtual link. The other endpoint of the virtual link is an FC switch (or FCF) port. A VN_Port emulates a native FC N_Port and performs similar functions: handling the creation, detection, and flow of messages to and from the FC switch. A single ENode can host multiple VN_Ports. Each VN_Port has a separate, unique virtual link with a FC switch.

ENodes contain at least one lossless Ethernet media access controller (MAC). Each Ethernet MAC is paired with an FCoE controller. The lossless Ethernet MAC is a full-duplex Ethernet MAC that implements Ethernet extensions to avoid frame loss due to congestion and supports frames of at least 2500 bytes. The FCoE controller instantiates and terminates VN_Port instances dynamically as they are needed for FCoE sessions. Each VN_Port instance has a unique virtual link to an FC switch.



NOTE: A *session* is a fabric login (FLOGI) or fabric discovery (FDISC) login to the FC SAN fabric. Session does not refer to end-to-end server-to-storage sessions.

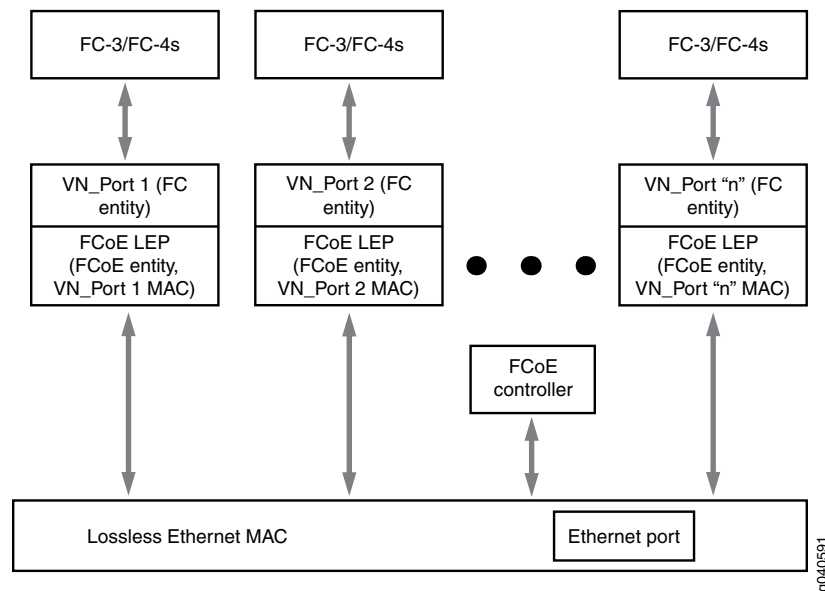
ENodes also contain one FCoE link end point (LEP) for each VN_Port connection. An FCoE LEP is a virtual FC interface mapped onto the physical Ethernet interface.

An FCoE LEP:

- Transmits and receives FCoE frames on the virtual link.
- Handles FC frame encapsulation for traffic going from the server to the FC switch.
- Performs frame de-encapsulation of traffic received from the FC switch.

Figure 2 on page 28 shows a block diagram of the major ENode components.

Figure 2: ENode Components



FCoE Frames

The FCoE protocol specification replaces the FC0 and FC1 layers of the FC stack with Ethernet, but retains the FC frame header. Retaining the FC frame header enables the FC frame to pass directly to a native FC SAN after de-encapsulation. The FCoE header carries the FC start of file (SOF) bits and end of file (EOF) bits in an encoded format. FCoE supports two frame types, control frames and data frames. FCoE Initialization Protocol (FIP) carries all of the discovery and fabric login frames.

FIP control frames handle FCoE device discovery, initializing communication, and maintaining communication. They do not carry a data payload. FIP has its own EtherType (0x8914) to distinguish FIP traffic from FCoE traffic and other Ethernet traffic. To establish communication, the ENode uses the globally unique MAC address assigned to it by the CNA manufacturer.

After FIP establishes a connection between FCoE devices, the FCoE data frames handle the transport of the FC frames encapsulated in Ethernet. FCoE also has its own EtherType (0x8906) to distinguish FCoE frames from other Ethernet traffic and ensure the in-order frame handling that FC requires. FCoE frames include:

- 2112 bytes FC payload
- 24 bytes FC header
- 14 bytes standard Ethernet header
- 14 bytes FCoE header
- 8 bytes cyclic redundancy check (CRC) plus EOF
- 4 bytes VLAN header
- 4 bytes frame check sequence (FCS)

The payload, headers, and checks add up to 2180 bytes. Therefore, interfaces that carry FCoE traffic should have a configured maximum transmission unit (MTU) of 2180 or larger. An MTU size of 2180 bytes is the minimum size; some network administrators prefer an MTU of 2240 or 2500 bytes.

Virtual Links

Native FC uses point-to-point physical links between FC devices. In FCoE, virtual links replace the physical links. A virtual link emulates a point-to-point link between two FCoE device endpoints, such as a server VN_Port and an FC switch (or FCF) VF_Port.

Each FCoE interface can support multiple virtual links. The MAC addresses of the FCoE endpoints (the VN_Port and the VF_Port) uniquely identify each virtual link and allow traffic for multiple virtual links to share the same physical link while maintaining data separation and security.

A virtual link exists in one FCoE VLAN and cannot belong to more than one VLAN. Although the FC switch and the FCoE device detect a virtual link as a point-to-point connection, virtual links do not need to be direct connections between a VF_Port and a VN_Port. A virtual link can traverse one or more transit switches, also known as passthrough switches. A transit switch can transparently aggregate virtual links while still appearing and functioning as a point-to-point connection to the FCoE devices. However, a virtual link must remain within a single Layer 2 domain.

FCoE VLANs

All FCoE traffic must travel in a VLAN dedicated to transporting only FCoE traffic. Only FCoE interfaces should be members of an FCoE VLAN. Ethernet traffic that is not FCoE or FIP traffic must travel in a different VLAN.



.....

NOTE: On a standalone switch or QFabric system Node device, the same VLAN cannot be used in both transit switch mode and FCoE-FC gateway mode.

.....



NOTE: FCoE VLANs (any VLAN that carries FCoE traffic) support only Spanning Tree Protocol (STP) and link aggregation group (LAG) Layer 2 features.

FCoE traffic cannot use a standard LAG because traffic might be hashed to different physical LAG links on different transmissions. This breaks the (virtual) point-to-point link that Fibre Channel traffic requires. If you configure a standard LAG interface for FCoE traffic, FCoE traffic might be rejected by the FC SAN.

QFabric systems support a special LAG called an FCoE LAG, which enables you to transport FCoE traffic and regular Ethernet traffic (traffic that is not FCoE traffic) across the same link aggregation bundle. Standard LAGs use a hashing algorithm to determine which physical link in the LAG is used for a transmission, so communication between two devices might use different physical links in the LAG for different transmissions. An FCoE LAG ensures that FCoE traffic uses the same physical link in the LAG for requests and replies in order to preserve the virtual point-to-point link between the FCoE device converged network adapter (CNA) and the FC SAN switch across the QFabric system Node device. An FCoE LAG does not provide load balancing or link redundancy for FCoE traffic. However, regular Ethernet traffic uses the standard hashing algorithm and receives the usual LAG benefits of load balancing and link redundancy in an FCoE LAG.



NOTE: IGMP snooping is enabled by default on all VLANs in all software versions before Junos OS R13.2. Disable IGMP snooping on FCoE VLANs if you are using software that is older than 13.2.

You can configure more than one FCoE VLAN, but any given virtual link must be in only one FCoE VLAN.



NOTE: All 10-Gigabit Ethernet interfaces that connect to FCoE devices must have a native VLAN configured in order to transport FIP traffic, because FIP VLAN discovery and notification frames are exchanged as untagged packets.

On switches that use the Enhanced Layer 2 Software (ELS) CLI, it is not sufficient only to configure the native VLAN on the interface, the interface must also be configured as a member of the native VLAN. (This is because the ELS CLI does not support tagged-access interface mode, so interfaces that are members of FCoE VLANs must use trunk mode, and trunk port interfaces must be explicitly included as members of a native VLAN.)

In addition, the VLAN ID must match the native VLAN ID that you configure on the physical interface. For example, to configure a native VLAN with an ID of 20 on interface xe-0/0/15 that is a member of an FCoE VLAN, you must include both of the following statements in the configuration:

1. Configure the native VLAN on the interface:

```
user@switch# set interfaces xe-0/0/15 native-vlan-id 20
```

(The equivalent configuration statement on a non-ELS device switch would be `set interfaces xe-0/0/15 unit 0 family ethernet-switching native-vlan-id 20`.)

2. Configure the port as a member of the native VLAN (this step is not required on switches that do not use the ELS software):

```
user@switch# set interfaces xe-0/0/15 unit 0 family ethernet-switching vlan members 20
```



BEST PRACTICE: Only FCoE traffic is permitted on the FCoE VLAN. A native VLAN might need to carry untagged traffic of different types and protocols. Therefore, it is a good practice to keep the native VLAN separate from FCoE VLANs.

Related Documentation

- [Understanding DCB Features and Requirements on page 78](#)
- [Understanding FCoE Transit Switch Functionality on page 22](#)
- [Understanding CoS Flow Control \(Ethernet PAUSE and PFC\) on page 81](#)
- [Configuring VLANs for FCoE Traffic on an FCoE Transit Switch on page 32](#)
- [Example: Configuring CoS PFC for FCoE Traffic on page 92](#)

Configuring VLANs for FCoE Traffic on an FCoE Transit Switch

When you configure a switch as a Fibre Channel over Ethernet (FCoE) transit switch, you must configure a VLAN that transports only FCoE traffic. FCoE traffic requires a dedicated VLAN and cannot share a VLAN with any other type of traffic. Because FCoE traffic is tagged traffic, the port (or interface) mode cannot be access mode, it must be either tagged-access port-mode (for switches that run the original CLI) or trunk interface-mode (for switches that run the Enhanced Layer 2 Software (ELS) CLI).

However, each interface that belongs to an FCoE VLAN must not only transport the tagged FCoE traffic, it must also transport the untagged FCoE Initialization Protocol (FIP) traffic. FIP communicates with the storage area network (SAN) Fibre Channel (FC) switch to set up the FCoE session for the FCoE client.

To transport untagged traffic on a tagged-access or trunk mode interface, the interface must have a native VLAN configured on it. Therefore, each interface that belongs to an FCoE VLAN must also have a native VLAN on it.

There are slight differences in the way you configure a native VLAN on an interface, depending on whether the switch uses the ELS CLI or the original CLI. This topic describes both methods.



NOTE: FCoE VLANs (any VLAN that carries FCoE traffic) support only Spanning Tree Protocol (STP) and link aggregation group (LAG) Layer 2 features.

FCoE traffic cannot use a standard LAG because traffic might be hashed to different physical LAG links on different transmissions. This breaks the (virtual) point-to-point link that Fibre Channel traffic requires. If you configure a standard LAG interface for FCoE traffic, FCoE traffic might be rejected by the FC SAN.

QFabric systems support a special LAG called an FCoE LAG, which enables you to transport FCoE traffic and regular Ethernet traffic (traffic that is not FCoE traffic) across the same link aggregation bundle. Standard LAGs use a hashing algorithm to determine which physical link in the LAG is used for a transmission, so communication between two devices might use different physical links in the LAG for different transmissions. An FCoE LAG ensures that FCoE traffic uses the same physical link in the LAG for requests and replies in order to preserve the virtual point-to-point link between the FCoE device converged network adapter (CNA) and the FC SAN switch across the QFabric system Node device. An FCoE LAG does not provide load balancing or link redundancy for FCoE traffic. However, regular Ethernet traffic uses the standard hashing algorithm and receives the usual LAG benefits of load balancing and link redundancy in an FCoE LAG.



NOTE: To configure an FCoE VLAN on a QFX3500 switch that you are using as an FCoE-FC gateway, you must also configure an FCoE VLAN interface as described in *Configuring an FCoE VLAN Interface on an FCoE-FC Gateway*. (Only the QFX3500 switch supports FCoE-FC gateway configuration.)

FCoE VLAN configuration includes:

- Configuring a VLAN to use as a dedicated FCoE VLAN
- Configuring the interface members of the FCoE VLAN.
- Configuring a native VLAN for FIP traffic.

This topic includes two configuration procedures, one for switches that run the original CLI, and one for switches that run the ELS CLI.

Original CLI Configuration

To configure an FCoE VLAN on a non-ELS switch:

1. Configure a dedicated FCoE VLAN:

```
[edit vlans]
user@switch# set vlan-name vlan-id vlan-id
```

For example, to configure a VLAN named **fcoe_vlan** with a VLAN ID of **100** as the FCoE VLAN:

```
[edit vlans]
user@switch# set fcoe_vlan vlan-id 100
```

2. Configure the FCoE VLAN on the interface (use **ethernet-switching** as the family and **tagged-access** as the port mode):

```
[edit interfaces]
user@switch# set interface-name unit unit family family port-mode mode vlan members
vlan-name
```

For example, to configure the interface **xe-0/0/10** as a member of the FCoE VLAN **fcoe_vlan**:

```
[edit interfaces]
user@switch# set xe-0/0/10 unit 0 family ethernet-switching port-mode tagged-access vlan
members fcoe_vlan
```

3. Configure the Ethernet interface membership in the FCoE VLAN:

```
[edit vlans]
user@switch# set vlan-name interface interface-name
```

For example, to assign the interface **xe-0/0/10.0** to the FCoE VLAN named **fcoe_vlan**:

```
[edit vlans]
user@switch# set fcoe_vlan interface xe-0/0/10.0
```

4. Configure a native VLAN for the untagged FIP traffic:

```
[edit vlans]
user@switch# set native vlan-id vlan-id
```

For example, to configure the native VLAN with a VLAN ID of 1:

```
[edit vlans]
user@switch# set native vlan-id 1
```

5. Assign member interfaces to the native VLAN:

```
[edit interfaces]
user@switch# set interface-name unit unit family family native-vlan-id vlan-id
```

For example, to configure the interface **xe-0/0/10** as a member of the native VLAN with the native VLAN ID 1:

```
[edit interfaces]
user@switch# set xe-0/0/10 unit 0 family ethernet-switching native-vlan-id 1
```

ELS CLI Configuration

To configure an FCoE VLAN on a switch running ELS:

1. Configure a dedicated FCoE VLAN:

```
[edit vlans]
user@switch# set vlan-name vlan-id vlan-id
```

For example, to configure a VLAN named **fcoe_vlan** with a VLAN ID of **100** as the FCoE VLAN:

```
[edit vlans]
user@switch# set fcoe_vlan vlan-id 100
```

2. Configure the FCoE VLAN on the interface (use **ethernet-switching** as the family and **trunk** as the interface mode):

```
[edit interfaces]
user@switch# set interface-name unit unit family family interface-mode mode vlan members
vlan-name
```

For example, to configure the interface **xe-0/0/10** as a member of the FCoE VLAN **fcoe_vlan**:

```
[edit interfaces]
user@switch# set xe-0/0/10 unit 0 family ethernet-switching interface-mode trunk vlan
members fcoe_vlan
```

3. Configure the Ethernet interface membership in the FCoE VLAN:

```
[edit vlans]
user@switch# set vlan-name interface interface-name
```

For example, to assign the interface **xe-0/0/10.0** to the FCoE VLAN named **fcoe_vlan**:

```
[edit vlans]
user@switch# set fcoe_vlan interface xe-0/0/10.0
```

4. Configure a native VLAN on the physical Ethernet interface for the untagged FIP traffic:

```
[edit interfaces]
user@switch# set interface-name native-vlan-id vlan-id
```

For example, to configure the native VLAN on interface **xe-0/0/10** with a VLAN ID of **1**:

```
[edit interfaces]
user@switch# set xe-0/0/10 native-vlan-id 1
```

5. Configure the Ethernet interface as a member of the native VLAN:

```
[edit interfaces]
user@switch# set interface-name unit unit family family vlan members native-vlan-id
```



NOTE: The *native-vlan-id* number must be the same as the native VLAN ID number that you configured on the physical Ethernet interface (see step 4).

For example, to configure the interface **xe-0/0/10** as a member of the native VLAN with the native VLAN ID **1**:

```
[edit interfaces]
```

```
user@switch# set xe-0/0/10 unit 0 family ethernet-switching vlan members 1
```

Related Documentation

- [Understanding FCoE on page 26](#)
- [Understanding FCoE Transit Switch Functionality on page 22](#)
- [Example: Configuring CoS PFC for FCoE Traffic on page 92](#)

Troubleshooting Dropped FCoE Traffic

Problem **Description:** Fibre Channel over Ethernet (FCoE) traffic for which you want guaranteed delivery is dropped.

Cause There are several possible causes of dropped FCoE traffic (the list numbers of the possible causes correspond to the list numbers of the solutions in the *Solution* section.):

1. Priority-based flow control (PFC) is not enabled on the FCoE priority (IEEE 802.1p code point) in both the input and output stanzas of the congestion notification profile.
2. The FCoE traffic is not classified correctly at the ingress interface. FCoE traffic should either use the default **fcoe** forwarding class and classifier configuration (maps the **fcoe** forwarding class to IEEE 802.1p code point 011) or be mapped to a lossless forwarding class and to the code point enabled for PFC on the input and output interfaces.
3. The congestion notification profile that enables PFC on the FCoE priority is not attached to the interface.
4. The forwarding class set (priority group) used for guaranteed delivery traffic does not include the forwarding class used for FCoE traffic.



NOTE: This issue can occur only on switches that support enhanced transmission selection (ETS) hierarchical port scheduling. (Direct port scheduling does not use forwarding class sets.)

5. Insufficient bandwidth has been allocated for the FCoE queue or for the forwarding class set to which the FCoE queue belongs.



NOTE: This issue can occur for forwarding class sets only on switches that support ETS hierarchical port scheduling. (Direct port scheduling does not use forwarding class sets.)

6. If you are using Junos OS Release 12.2, the **fcoe** forwarding class has been explicitly configured instead of using the default **fcoe** forwarding class configuration (forwarding-class-to-queue mapping).



NOTE: If you are using Junos OS Release 12.2, use the default forwarding-class-to-queue mapping for the lossless fcoe and no-loss forwarding classes. If you explicitly configure the lossless forwarding classes, the traffic mapped to those forwarding classes is treated as lossy (best effort) traffic and does *not* receive lossless treatment.

7. If you are using Junos OS Release 12.3 or later and you are not using the default **fcoe** forwarding class configuration, the forwarding class used for FCoE is not configured with the **no-loss** packet drop attribute. In Junos OS 12.3 or later, explicit forwarding classes configurations must include the **no-loss** packet drop attribute to be treated as lossless forwarding classes.

Solution The list numbers of the possible solutions correspond to the list numbers of the causes in the *Cause* section.

1. Check the congestion notification profile (CNP) to see if PFC is enabled on the FCoE priority (the correct IEEE 802.1p code point) on both input and output interfaces. Use the **show class-of-service congestion-notification** operational command to show the code points that are enabled for PFC in each CNP.

If you are using the default configuration, FCoE traffic is mapped to code point 011 (priority 3). In this case, the input stanza of the CNP should show that PFC is enabled on code point 011, and the output stanza should show that priority 011 is mapped to flow control queue 3.

If you explicitly configured a forwarding class for FCoE traffic, ensure that:

- You specified the **no-loss** packet drop attribute in the forwarding class configuration
- The code point mapped to the FCoE forwarding class in the ingress classifier is the code point enabled for PFC in the CNP input stanza
- The code point and output queue used for FCoE traffic are mapped to each other in the CNP output stanza (if you are not using the default priority and queue, you must explicitly configure each output queue that you want to respond to PFC messages)

For example, if you explicitly configure a forwarding class for FCoE traffic that is mapped to output queue 5 and to code point 101 (priority 5), the output of the **show class-of-service congestion-notification** looks like:

```
Name: fcoe_p5_cnp, Index: 12183
Type: Input
Cable Length: 100 m
  Priority  PFC      MRU
  000      Disabled
  001      Disabled
  010      Disabled
  011      Disabled
  100      Disabled
  101      Enabled   2500
  110      Disabled
  111      Disabled
Type: Output
  Priority  Flow-Control-Queues
  101      5
```

2. Use the **show class-of-service classifier type ieee-802.1p** operational command to check if the classifier maps the forwarding class used for FCoE traffic to the correct IEEE 802.1p code point.
3. Ensure that the congestion notification profile and classifier are attached to the correct ingress interface. Use the operational command **show configuration class-of-service interfaces interface-name**.
4. Check that the forwarding class set includes the forwarding class used for FCoE traffic. Use the operational command **show configuration class-of-service forwarding-class-sets** to show the configured priority groups and their forwarding classes.

5. Verify the amount of bandwidth allocated to the queue mapped to the FCoE forwarding class and to the forwarding class set to which the FCoE traffic queue belongs. Use the **show configuration class-of-service schedulers *scheduler-name*** operational command (specify the scheduler for FCoE traffic as the *scheduler-name*) to see the minimum guaranteed bandwidth (**transmit-rate**) and maximum bandwidth (**shaping-rate**) for the queue.

Use the **show configuration class-of-service traffic-control-profiles *traffic-control-profile*** operational command (specify the traffic control profile used for FCoE traffic as the *traffic-control-profile*) to see the minimum guaranteed bandwidth (**guaranteed-rate**) and maximum bandwidth (**shaping-rate**) for the forwarding class set.

6. Delete the explicit FCoE forwarding-class-to-queue mapping so that the system uses the default FCoE forwarding-class-to-queue mapping. Include the **delete forwarding-classes class fcoe queue-num 3** statement at the **[edit class-of-service]** hierarchy level to remove the explicit configuration. The system then uses the default configuration for the FCoE forwarding class and preserves the lossless treatment of FCoE traffic.
7. Use the **show class-of-service forwarding-class** operational command to display the configured forwarding classes. The *No-Loss* column shows whether lossless transport is enabled or disabled for each forwarding class. If the forwarding class used for FCoE traffic is not enabled for lossless transport, include the **no-loss** packet drop attribute in the forwarding class configuration (**set class-of-service forwarding-classes class *fcoe-forwarding-class-name* queue-num *queue-number* no-loss**).

See [“Example: Configuring CoS PFC for FCoE Traffic” on page 92](#) for step-by-step instructions on how to configure PFC for FCoE traffic, including classifier, interface, congestion notification profile, PFC, and bandwidth scheduling configuration.

Related Documentation

- *show class-of-service congestion-notification*
- *Configuring CoS PFC (Congestion Notification Profiles)*
- [Example: Configuring CoS PFC for FCoE Traffic on page 92](#)
- [Understanding CoS Flow Control \(Ethernet PAUSE and PFC\) on page 81](#)

Understanding MC-LAGs on an FCoE Transit Switch

Multichassis link aggregation groups (MC-LAGs) provide redundancy and load balancing between two switches, multihoming support for client devices such as servers, and a loop-free Layer 2 network without running Spanning Tree Protocol (STP).

You can use an MC-LAG to provide a redundant aggregation layer for Fibre Channel over Ethernet (FCoE) traffic. To support lossless transport of FCoE traffic across an MC-LAG, you must configure the appropriate class of service (CoS) on both of the switches with MC-LAG port members. The CoS configuration must be the same on both of the MC-LAG switches because MC-LAGs do not carry forwarding class and IEEE 802.1p priority information.

Ports that are part of an FCoE-FC gateway configuration (a virtual FCoE-FC gateway fabric) do not support MC-LAGs. Ports that are members of an MC-LAG act as pass-through transit switch ports.

Standalone switches support MC-LAGs. QFabric system Node devices do not support MC-LAGs. Virtual Chassis and mixed-mode Virtual Chassis Fabric (VCF) configurations do not support FCoE. Only pure QFX5100 VCFs (consisting of only QFX5100 switches) support FCoE.

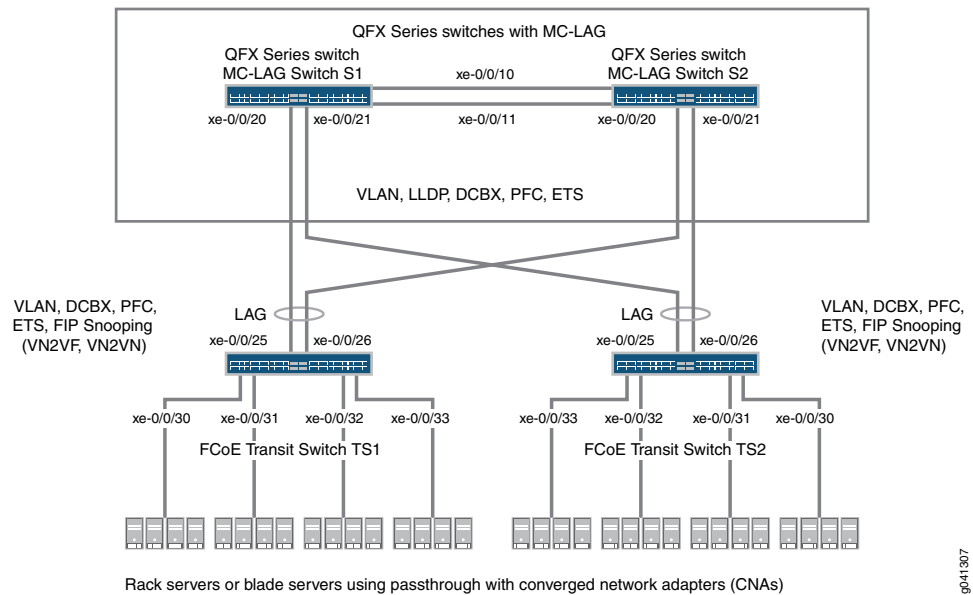
This topic describes:

- [Supported Topology on page 41](#)
- [FIP Snooping and FCoE Trusted Ports on page 43](#)
- [CoS and Data Center Bridging \(DCB\) on page 44](#)

Supported Topology

Switches that are not directly connected to FCoE hosts and that act as pass-through transit switches support MC-LAGs for FCoE traffic in an *inverted-U* network topology. [Figure 3 on page 42](#) shows an inverted-U topology using QFX3500 switches.

Figure 3: Supported Topology for an MC-LAG on an FCoE Transit Switch



The following rules and guidelines apply to MC-LAGs when used for FCoE traffic. The rules and guidelines help to ensure the proper handling and lossless transport characteristics required for FCoE traffic.

- The two switches that form the MC-LAG (Switches S1 and S2) cannot use ports that are part of an FCoE-FC gateway fabric. The MC-LAG switch ports must be pass-through transit switch ports (used as part of an intermediate transit switch that is not directly connected to FCoE hosts).
- MC-LAG Switches S1 and S2 cannot be directly connected to the FCoE hosts.
- The two switches that serve as access devices for FCoE hosts (FCoE Transit Switches TS1 and TS2) use standard LAGs to connect to MC-LAG Switches S1 and S2. FCoE Transit Switches TS1 and TS2 can be standalone switches or they can be Node devices in a QFabric system.
- Transit Switches TS1 and TS2 must use transit switch ports for the FCoE hosts and for the standard LAGs to MC-LAG Switches S1 and S2.
- Enable FIP snooping on the FCoE VLAN on Transit Switches TS1 and TS2. You can configure either VN_Port to VF_Port (VN2VF_Port) FIP snooping or VN_Port to VN_Port (VN2VN_Port) FIP snooping, depending on whether the FCoE hosts need to access targets in the FC SAN (VN2VF_Port FIP snooping) or targets in the Ethernet network (VN2VN_Port FIP snooping).

FIP snooping should be performed at the access edge and is not supported on MC-LAG switches. Do not enable FIP snooping on MC-LAG Switches S1 and S2. (Do not enable FIP snooping on the MC-LAG ports that connect Switches S1 and S2 to Switches TS1 and TS2 or on the LAG ports that connect Switch S1 to S2.)



NOTE: Juniper Networks QFX10000 aggregation switches do not support FIP snooping, so they cannot be used as FIP snooping access switches (Transit Switches TS1 and TS2) in this topology.

- The CoS configuration must be consistent on the MC-LAG switches. Because MC-LAGs carry no forwarding class or priority information, each MC-LAG switch needs to have the same CoS configuration to support lossless transport. (On each MC-LAG switch, the name, egress queue, and CoS provisioning of each forwarding class must be the same, and the priority-based flow control (PFC) configuration must be the same.)

Transit Switches (Server Access)

The role of FCoE Transit Switches TS1 and TS2 is to connect FCoE hosts in a multihomed fashion to the MC-LAG switches, so Transit Switches TS1 and TS2 act as access switches for the FCoE hosts. (FCoE hosts are directly connected to Transit Switches TS1 and TS2.)

The transit switch configuration depends on whether you want to do VN2VF_Port FIP snooping or VN2VN_Port FIP snooping, and whether the transit switches also have ports configured as part of an FCoE-FC gateway virtual fabric. Ports that a QFX3500 switch uses in an FCoE-FC gateway virtual fabric cannot be included in the transit switch LAG connection to the MC-LAG switches. (Ports cannot belong to both a transit switch and an FCoE-FC gateway; you must use different ports for each mode of operation.)

MC-LAG Switches (FCoE Aggregation)

The role of MC-LAG Switches S1 and S2 is to provide redundant, load-balanced connections between FCoE transit switches. The MC-LAG Switches S1 and S2 act as aggregation switches. FCoE hosts are not directly connected to the MC-LAG switches.

The MC-LAG switch configuration is the same regardless of which type of FIP snooping FCoE Transit Switches TS1 and TS2 perform.

FIP Snooping and FCoE Trusted Ports

To maintain secure access, enable VN2VF_Port FIP snooping or VN2VN_Port FIP snooping at the transit switch access ports connected directly to the FCoE hosts. FIP snooping should be performed at the access edge of the network to prevent unauthorized access. For example, in [Figure 3 on page 42](#), you enable FIP snooping on the FCoE VLANs on Transit Switches TS1 and TS2 that include the access ports connected to the FCoE hosts.

Do not enable FIP snooping on the switches used to create the MC-LAG. For example, in [Figure 3 on page 42](#), you would not enable FIP snooping on the FCoE VLANs on Switches S1 and S2.

Configure links between switches as FCoE trusted ports to reduce FIP snooping overhead and ensure that the system performs FIP snooping only at the access edge. In the sample topology, configure the Transit Switch TS1 and TS2 LAG ports connected to the MC-LAG switches as FCoE trusted ports, configure the Switch S1 and S2 MC-LAG ports connected to Switches TS1 and TS2 as FCoE trusted ports, and configure the ports in the LAG that connects Switches S1 to S2 as FCoE trusted ports.

CoS and Data Center Bridging (DCB)

The MC-LAG links do not carry forwarding class or priority information. The following CoS properties must have the same configuration on each MC-LAG switch or on each MC-LAG interface to support lossless transport:

- FCoE forwarding class name—For example, the forwarding class for FCoE traffic could use the default **fcoe** forwarding class on both MC-LAG switches.
- FCoE output queue—For example, the **fcoe** forwarding class could be mapped to queue 3 on both MC-LAG switches (queue 3 is the default mapping for the **fcoe** forwarding class).
- Classifier—The forwarding class for FCoE traffic must be mapped to the same IEEE 802.1p code point on each member interface of the MC-LAG on both MC-LAG switches. For example, the FCoE forwarding class **fcoe** could be mapped to IEEE 802.1p code point **011** (code point **011** is the default mapping for the **fcoe** forwarding class).
- Priority-based flow control (PFC)—PFC must be enabled on the FCoE code point on each MC-LAG switch and applied to each MC-LAG interface using a congestion notification profile.

You must also configure enhanced transmission selection (ETS) on the MC-LAG interfaces to provide sufficient scheduling resources (bandwidth, priority) for lossless transport. The ETS configuration can be different on each MC-LAG switch, as long as enough resources are scheduled to support lossless transport for the expected FCoE traffic.

Link Layer Discovery Protocol (LLDP) and Data Center Bridging Capability Exchange Protocol (DCBX) must be enabled on each MC-LAG member interface (LLDP and DCBX are enabled by default on all interfaces).



NOTE: As with all other FCoE configurations, FCoE traffic requires a dedicated VLAN that carries only FCoE traffic, and IGMP snooping must be disabled on the FCoE VLAN.

Example: Configuring CoS for FCoE Transit Switch Traffic Across an MC-LAG

Multichassis link aggregation groups (MC-LAGs) provide redundancy and load balancing between two QFX Series switches, multihoming support for client devices such as servers, and a loop-free Layer 2 network without running Spanning Tree Protocol (STP).



NOTE: This example uses the Junos OS Enhanced Layer 2 Software (ELS) configuration style for QFX Series switches. If your switch runs software that does not support ELS, see *Example: Configuring CoS for FCoE Transit Switch Traffic Across an MC-LAG*. For ELS details, see *Getting Started with Enhanced Layer 2 Software*.

You can use an MC-LAG to provide a redundant aggregation layer for Fibre Channel over Ethernet (FCoE) traffic in an *inverted-U* topology. To support lossless transport of FCoE traffic across an MC-LAG, you must configure the appropriate class of service (CoS) on both of the QFX Series switches with MC-LAG port members. The CoS configuration must be the same on both of the MC-LAG switches because an MC-LAG does not carry forwarding class and IEEE 802.1p priority information.

Ports that are members of an MC-LAG act as FCoE passthrough transit switch ports.



NOTE: This example describes how to configure CoS to provide lossless transport for FCoE traffic across an MC-LAG that connects two QFX Series switches. It also describes how to configure CoS on the FCoE transit switches that connect FCoE hosts to the QFX Series switches that form the MC-LAG.

This example does *not* describe how to configure the MC-LAG itself. For a detailed example of MC-LAG configuration, see *Example: Configuring Multichassis Link Aggregation*. However, this example includes a subset of MC-LAG configuration that only shows how to configure interface membership in the MC-LAG.



NOTE: Juniper Networks QFX10000 aggregation switches do not support FIP snooping, so they cannot be used as FIP snooping access switches (Transit Switches TS1 and TS2) in this example. However, QFX10000 switches can play the role of the MC-LAG switches (MC-LAG Switch S1 and MC-LAG Switch S2) in this example.

QFX3500 and QFX3600 Virtual Chassis switches do not support FCoE.

This topic describes:

- [Requirements on page 45](#)
- [Overview on page 46](#)
- [Configuration on page 51](#)
- [Verification on page 61](#)

Requirements

This example uses the following hardware and software components:

- Two Juniper Networks QFX5100 Switches running the ELS CLI that form an MC-LAG for FCoE traffic.
- Two Juniper Networks QFX5100 Switches running the ELS CLI that provide FCoE server access in transit switch mode and that connect to the MC-LAG switches.
- FCoE servers (or other FCoE hosts) connected to the transit switches.
- Junos OS Release 13.2 or later for the QFX Series.

Overview

FCoE traffic requires lossless transport. This example shows you how to:

- Configure CoS for FCoE traffic on the two QFX5100 switches that form the MC-LAG, including priority-based flow control (PFC). The example also includes configuration for both enhanced transmission selection (ETS) hierarchical scheduling of resources for the FCoE forwarding class priority and for the forwarding class set priority group, and also direct port scheduling. You can only use one of the scheduling methods on a port. Different switches support different scheduling methods.



NOTE: Configuring or changing PFC on an interface blocks the entire port until the PFC change is completed. After a PFC change is completed, the port is unblocked and traffic resumes. Blocking the port stops ingress and egress traffic, and causes packet loss on all queues on the port until the port is unblocked.

- Configure CoS for FCoE on the two FCoE transit switches that connect FCoE hosts to the MC-LAG switches and enable FIP snooping on the FCoE VLAN at the FCoE transit switch access ports.
- Configure the appropriate port mode, MTU, and FCoE trusted or untrusted state for each interface to support lossless FCoE transport.

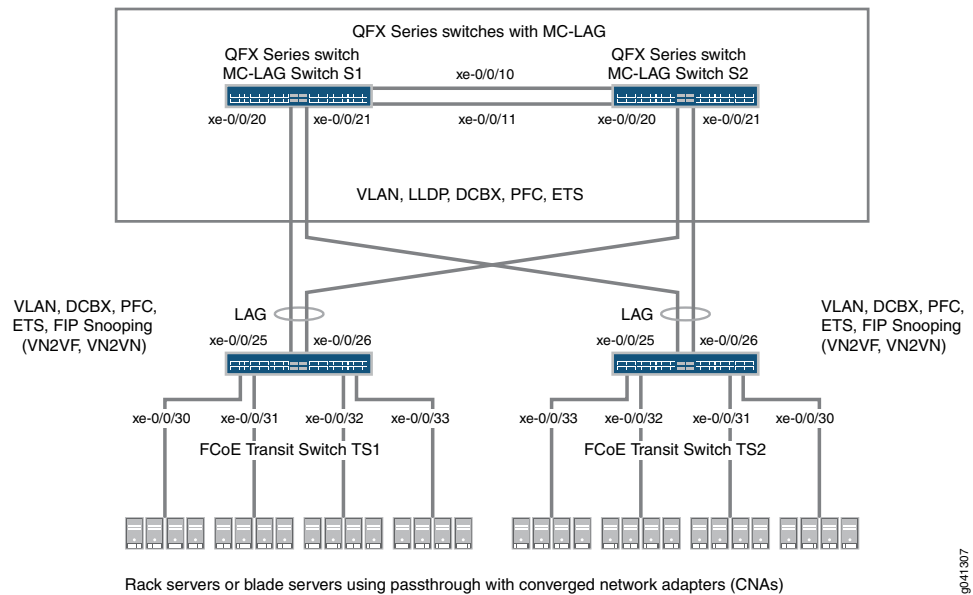


NOTE: Do not enable IGMP snooping on the FCoE VLAN. (IGMP snooping is enabled on the default VLAN by default, but is disabled by default on all other VLANs.)

Topology

QFX5100 switches that act as transit switches support MC-LAGs for FCoE traffic in an inverted-U network topology, as shown in [Figure 4 on page 47](#).

Figure 4: Supported Topology for an MC-LAG on an FCoE Transit Switch



NOTE: Juniper Networks QFX10000 aggregation switches do not support FIP snooping, so they cannot be used as FIP snooping access switches (Transit Switches TS1 and TS2) in this example. However, QFX10000 switches can play the role of the MC-LAG switches (MC-LAG Switch S1 and MC-LAG Switch S2) in this example.

Table 3 on page 47 shows the configuration components for this example.

Table 3: Components of the CoS for FCoE Traffic Across an MC-LAG Configuration Topology

Component	Settings
Hardware	Four QFX5100 switches running the ELS CLI (two to form the MC-LAG as passthrough transit switches and two transit switches for FCoE access).
Forwarding class (all switches)	Default fcoe forwarding class.
Classifier (forwarding class mapping of incoming traffic to IEEE priority)	Default IEEE 802.1p trusted classifier on all FCoE interfaces.

Table 3: Components of the CoS for FCoE Traffic Across an MC-LAG Configuration Topology (continued)

Component	Settings
LAGs and MC-LAG	<p>S1—Ports xe-0/0/10 and x-0/0/11 are members of LAG ae0, which connects Switch S1 to Switch S2. Ports xe-0/0/20 and xe-0/0/21 are members of MC-LAG ae1. All ports are configured in trunk interface mode, as fcoe-trusted, and with an MTU of 2180.</p> <p>S2—Ports xe-0/0/10 and x-0/0/11 are members of LAG ae0, which connects Switch S2 to Switch S1. Ports xe-0/0/20 and xe-0/0/21 are members of MC-LAG ae1. All ports are configured in trunk interface mode, as fcoe-trusted, and with an MTU of 2180.</p> <p>NOTE: Ports xe-0/0/20 and xe-0/0/21 on Switches S1 and S2 are the members of the MC-LAG.</p> <p>TS1—Ports xe-0/0/25 and x-0/0/26 are members of LAG ae1, configured in trunk interface mode, as fcoe-trusted, and with an MTU of 2180. Ports xe-0/0/30, xe-0/0/31, xe-0/0/32, and xe-0/0/33 are configured in trunk interface mode, with an MTU of 2180.</p> <p>TS2—Ports xe-0/0/25 and x-0/0/26 are members of LAG ae1, configured in trunk interface mode, as fcoe-trusted, and with an MTU of 2180. Ports xe-0/0/30, xe-0/0/31, xe-0/0/32, and xe-0/0/33 are configured in trunk interface mode, with an MTU of 2180.</p>
FCoE queue scheduler (all switches)	<p>fcoe-sched: Minimum bandwidth 3g Maximum bandwidth 100% Priority low</p>
Forwarding class-to-scheduler mapping (all switches)	<p>Scheduler map fcoe-map: Forwarding class fcoe Scheduler fcoe-sched</p> <p>NOTE: If you are using direct port scheduling,</p>
PFC congestion notification profile (all switches)	<p>fcoe-cnp: Code point 011</p> <p>Ingress interfaces:</p> <ul style="list-style-type: none"> • S1—LAG ae0 and MC-LAG ae1 • S2—LAG ae0 and MC-LAG ae1 • TS1—LAG ae1, interfaces xe-0/0/30, xe-0/0/31, xe-0/0/32, and xe-0/0/33 • TS2—LAG ae1, interfaces xe-0/0/30, xe-0/0/31, xe-0/0/32, and xe-0/0/33

Table 3: Components of the CoS for FCoE Traffic Across an MC-LAG Configuration Topology (*continued*)

Component	Settings
FCoE VLAN name and tag ID	<p>Name—fcoe_vlan ID—100</p> <p>Include the FCoE VLAN on the interfaces that carry FCoE traffic on all four switches.</p>
ETS only—forwarding class set (FCoE priority group, all switches)	<p>fcoe-pg: Forwarding class fcoe</p> <p>Egress interfaces:</p> <ul style="list-style-type: none"> • S1—LAG ae0 and MC-LAG ae1 • S2—LAG ae0 and MC-LAG ae1 • TS1—LAG ae1, interfaces xe-0/0/30, xe-0/0/31, xe-0/0/32, and xe-0/0/33 • TS2—LAG ae1, interfaces xe-0/0/30, xe-0/0/31, xe-0/0/32, and xe-0/0/33
ETS only—traffic control profile (all switches)	<p>fcoe-tcp: Scheduler map fcoe-map Minimum bandwidth 3g Maximum bandwidth 100%</p> <p>The traffic control profile is applied to the same interfaces as the forwarding class set, using the same CLI statement. This applies ETS hierarchical scheduling to the interfaces.</p>
Port scheduling only—apply scheduling to interfaces	<p>On switches that support direct port scheduling, if you use port scheduling, apply scheduling by attaching the scheduler map directly to interfaces:</p> <ul style="list-style-type: none"> • S1—LAG ae0 and MC-LAG ae1 • S2—LAG ae0 and MC-LAG ae1 • TS1—LAG ae1, interfaces xe-0/0/30, xe-0/0/31, xe-0/0/32, and xe-0/0/33 • TS2—LAG ae1, interfaces xe-0/0/30, xe-0/0/31, xe-0/0/32, and xe-0/0/33
FIP snooping	<p>Enable FIP snooping on Transit Switches TS1 and TS2 on the FCoE VLAN. Configure the LAG interfaces that connect to the MC-LAG switches as FCoE trusted interfaces so that they do not perform FIP snooping.</p> <p>This example enables VN2VN_Port FIP snooping on the FCoE transit switch interfaces connected to the FCoE servers. The example is equally valid with VN2VF_Port FIP snooping enabled on the transit switch access ports. The method of FIP snooping you enable depends on your network configuration.</p> <p>NOTE: Juniper Networks QFX10000 aggregation switches do not support FIP snooping, so they cannot be used as FIP snooping access switches (Transit Switches TS1 and TS2) in this example.</p>



NOTE: This example uses the default IEEE 802.1p trusted BA classifier, which is automatically applied to trunk mode interfaces if you do not apply an explicitly configured classifier.

To configure CoS for FCoE traffic across an MC-LAG:

- Use the default FCoE forwarding class and forwarding-class-to-queue mapping (do not explicitly configure the FCoE forwarding class or output queue). The default FCoE forwarding class is **fcoe**, and the default output queue is **queue 3**.
- Use the default trusted BA classifier, which maps incoming packets to forwarding classes by the IEEE 802.1p code point (CoS priority) of the packet. The trusted classifier is the default classifier for interfaces in trunk interface mode. The default trusted classifier maps incoming packets with the IEEE 802.1p code point 3 (**011**) to the FCoE forwarding class. If you choose to configure the BA classifier instead of using the default classifier, you must ensure that FCoE traffic is classified into forwarding classes in exactly the same way on both MC-LAG switches. Using the default classifier ensures consistent classifier configuration on the MC-LAG ports.
- Configure a congestion notification profile that enables PFC on the FCoE code point (code point **011** in this example). The congestion notification profile configuration must be the same on both MC-LAG switches.
- Apply the congestion notification profile to the interfaces.
- Configure the interface mode, MTU, and FCoE trusted or untrusted state for each interface to support lossless FCoE transport.
- For ETS hierarchical port scheduling, configure ETS on the interfaces to provide the bandwidth required for lossless FCoE transport. Configuring ETS includes configuring bandwidth scheduling for the FCoE forwarding class, a forwarding class set (priority group) that includes the FCoE forwarding class, and a traffic control profile to assign bandwidth to the forwarding class set that includes FCoE traffic, and applying the traffic control profile and forwarding class set to interfaces..

On switches that support direct port scheduling, configure CoS properties on interfaces by applying scheduler maps directly to interfaces.

In addition, this example describes how to enable FIP snooping on the Transit Switch TS1 and TS2 ports that are connected to the FCoE servers. To provide secure access, FIP snooping must be enabled on the FCoE access ports.

This example focuses on the CoS configuration to support lossless FCoE transport across an MC-LAG. This example does not describe how to configure the properties of MC-LAGs and LAGs, although it does show you how to configure the port characteristics required to support lossless transport and how to assign interfaces to the MC-LAG and to the LAGs.

Before you configure CoS, configure:

- The MC-LAGs that connect Switches S1 and S2 to Switches TS1 and TS2. (*Example: Configuring Multichassis Link Aggregation* describes how to configure MC-LAGs.)

- The LAGs that connect the Transit Switches TS1 and TS2 to MC-LAG Switches S1 and S2. (*Configuring Link Aggregation* describes how to configure LAGs.)
- The LAG that connects Switch S1 to Switch S2.

Configuration

To configure CoS for lossless FCoE transport across an MC-LAG, perform these tasks:

- [MC-LAG Switches S1 and S2 Common Configuration \(Applies to ETS and Port Scheduling\) on page 53](#)
- [MC-LAG Switches S1 and S2 ETS Hierarchical Scheduling Configuration on page 54](#)
- [MC-LAG Switches S1 and S2 Port Scheduling Configuration on page 55](#)
- [FCoE Transit Switches TS1 and TS2 Common Configuration \(Applies to ETS and Port Scheduling\) on page 55](#)
- [FCoE Transit Switches TS1 and TS2 ETS Hierarchical Scheduling Configuration on page 57](#)
- [FCoE Transit Switches TS1 and TS2 Port Scheduling Configuration on page 57](#)
- [Results on page 58](#)

CLI Quick Configuration



NOTE: The CLI configurations for the MC-LAG switches and for the FCoE transit switches are each separated into three sections:

- Configuration common to all port scheduling methods
- Configuration specific to ETS hierarchical port scheduling
- Configuration specific to direct port scheduling

MC-LAG Switch S1 and Switch S2

To quickly configure CoS for lossless FCoE transport across an MC-LAG, copy the following commands, paste them in a text file, remove line breaks, change variables and details to match your network configuration, and then copy and paste the commands into the CLI for MC-LAG Switch S1 and MC-LAG Switch S2 at the **[edit]** hierarchy level. The configurations on Switches S1 and S2 are identical because the CoS configuration must be identical, and because this example uses the same ports on both switches.

MC-LAG Switches Configuration Common to ETS Hierarchical Port Scheduling and to Direct Port Scheduling

```
set class-of-service schedulers fcoe-sched priority low transmit-rate 3g
set class-of-service schedulers fcoe-sched shaping-rate percent 100
set class-of-service scheduler-maps fcoe-map forwarding-class fcoe scheduler fcoe-sched
set class-of-service congestion-notification-profile fcoe-cnp input ieee-802.1 code-point 011 pfc
set class-of-service interfaces ae0 congestion-notification-profile fcoe-cnp
set class-of-service interfaces ae1 congestion-notification-profile fcoe-cnp
set vlans fcoe_vlan vlan-id 100
set interfaces xe-0/0/10 ether-options 802.3ad ae0
set interfaces xe-0/0/11 ether-options 802.3ad ae0
set interfaces xe-0/0/20 ether-options 802.3ad ae1
set interfaces xe-0/0/21 ether-options 802.3ad ae1
set interfaces ae0 unit 0 family ethernet-switching interface-mode trunk vlan members fcoe_vlan
set interfaces ae1 unit 0 family ethernet-switching interface-mode trunk vlan members fcoe_vlan
```

```
set interfaces ae0 mtu 2180
set interfaces ae1 mtu 2180
set vlans fcoe_vlan forwarding-options fip-security interface ae0 fcoe-trusted
set vlans fcoe_vlan forwarding-options fip-security interface ae1 fcoe-trusted
```

MC-LAG Switches Configuration for ETS Hierarchical Port Scheduling

```
set class-of-service forwarding-class-sets fcoe-pg class fcoe
set class-of-service traffic-control-profiles fcoe-tcp scheduler-map fcoe-map guaranteed-rate
3g
set class-of-service traffic-control-profiles fcoe-tcp shaping-rate percent 100
set class-of-service interfaces ae0 forwarding-class-set fcoe-pg output-traffic-control-profile
fcoe-tcp
set class-of-service interfaces ae1 forwarding-class-set fcoe-pg output-traffic-control-profile
fcoe-tcp
```

MC-LAG Switches Configuration for Direct Port Scheduling

```
set class-of-service interfaces ae0 scheduler-map fcoe-map
set class-of-service interfaces ae1 scheduler-map fcoe-map
```

FCoE Transit Switch TS1 and Switch TS2

To quickly configure CoS for lossless FCoE transport across an MC-LAG, copy the following commands, paste them in a text file, remove line breaks, change variables and details to match your network configuration, and then copy and paste the commands into the CLI for Transit Switch TS1 and Transit Switch TS2 at the **[edit]** hierarchy level. The configurations on Switches TS1 and TS2 are identical because the CoS configuration must be identical, and because this example uses the same ports on both switches.

FCoE Transit Switches Configuration Common to ETS Hierarchical Port Scheduling and to Direct Port Scheduling

```
set class-of-service schedulers fcoe-sched priority low transmit-rate 3g
set class-of-service schedulers fcoe-sched shaping-rate percent 100
set class-of-service scheduler-maps fcoe-map forwarding-class fcoe scheduler fcoe-sched
set class-of-service congestion-notification-profile fcoe-cnp input ieee-802.1 code-point 011 pfc
set class-of-service interfaces ae1 congestion-notification-profile fcoe-cnp
set class-of-service interfaces xe-0/0/30 congestion-notification-profile fcoe-cnp
set class-of-service interfaces xe-0/0/31 congestion-notification-profile fcoe-cnp
set class-of-service interfaces xe-0/0/32 congestion-notification-profile fcoe-cnp
set class-of-service interfaces xe-0/0/33 congestion-notification-profile fcoe-cnp
set vlans fcoe_vlan vlan-id 100
set interfaces xe-0/0/25 ether-options 802.3ad ae1
set interfaces xe-0/0/26 ether-options 802.3ad ae1
set interfaces ae1 unit 0 family ethernet-switching interface-mode trunk vlan members fcoe_vlan
set interfaces xe-0/0/30 unit 0 family ethernet-switching interface-mode trunk vlan members
fcoe_vlan
set interfaces xe-0/0/31 unit 0 family ethernet-switching interface-mode trunk vlan members
fcoe_vlan
set interfaces xe-0/0/32 unit 0 family ethernet-switching interface-mode trunk vlan members
fcoe_vlan
set interfaces xe-0/0/33 unit 0 family ethernet-switching interface-mode trunk vlan members
fcoe_vlan
set interfaces ae1 mtu 2180
set interfaces xe-0/0/30 mtu 2180
set interfaces xe-0/0/31 mtu 2180
set interfaces xe-0/0/32 mtu 2180
set interfaces xe-0/0/33 mtu 2180
set vlans fcoe_vlan forwarding-options fip-security interface ae1 fcoe-trusted
set vlans fcoe_vlan forwarding-options fip-security examine-vn2v2 beacon-period 90000
```

FCoE Transit Switches Configuration for ETS Hierarchical Port Scheduling

```

set class-of-service forwarding-class-sets fcoe-pg class fcoe
set class-of-service traffic-control-profiles fcoe-tcp scheduler-map fcoe-map guaranteed-rate
3g
set class-of-service traffic-control-profiles fcoe-tcp shaping-rate percent 100
set class-of-service interfaces ae1 forwarding-class-set fcoe-pg output-traffic-control-profile
fcoe-tcp
set class-of-service interfaces xe-0/0/30 forwarding-class-set fcoe-pg
output-traffic-control-profile fcoe-tcp
set class-of-service interfaces xe-0/0/31 forwarding-class-set fcoe-pg
output-traffic-control-profile fcoe-tcp
set class-of-service interfaces xe-0/0/32 forwarding-class-set fcoe-pg
output-traffic-control-profile fcoe-tcp
set class-of-service interfaces xe-0/0/33 forwarding-class-set fcoe-pg
output-traffic-control-profile fcoe-tcp

```

FCoE Transit Switches Configuration for Direct Port Scheduling

```

set class-of-service interfaces ae1 scheduler-map fcoe-map
set class-of-service interfaces xe-0/0/30 scheduler-map fcoe-map
set class-of-service interfaces xe-0/0/31 scheduler-map fcoe-map
set class-of-service interfaces xe-0/0/32 scheduler-map fcoe-map
set class-of-service interfaces xe-0/0/33 scheduler-map fcoe-map

```

MC-LAG Switches S1 and S2 Common Configuration (Applies to ETS and Port Scheduling)

Step-by-Step Procedure

To configure queue scheduling, PFC, the FCoE VLAN, and LAG and MC-LAG interface membership and characteristics to support lossless FCoE transport across an MC-LAG (this example uses the default **fcoe** forwarding class and the default classifier to map incoming FCoE traffic to the FCoE IEEE 802.1p code point 011), for both ETS hierarchical port scheduling and port scheduling (common configuration):

1. Configure output scheduling for the FCoE queue:


```

[edit class-of-service]
user@switch# set schedulers fcoe-sched priority low transmit-rate 3g
user@switch# set schedulers fcoe-sched shaping-rate percent 100

```
2. Map the FCoE forwarding class to the FCoE scheduler (**fcoe-sched**):


```

[edit class-of-service]
user@switch# set scheduler-maps fcoe-map forwarding-class fcoe scheduler fcoe-sched

```
3. Enable PFC on the FCoE priority by creating a congestion notification profile (**fcoe-cnp**) that applies FCoE to the IEEE 802.1 code point 011:


```

[edit class-of-service]
user@switch# set congestion-notification-profile fcoe-cnp input ieee-802.1 code-point
011 pfc

```
4. Apply the PFC configuration to the LAG and MC-LAG interfaces:


```

[edit class-of-service]
user@switch# set interfaces ae0 congestion-notification-profile fcoe-cnp
user@switch# set interfaces ae1 congestion-notification-profile fcoe-cnp

```
5. Configure the VLAN for FCoE traffic (**fcoe_vlan**):

- ```
[edit vlans]
user@switch# set fcoe_vlan vlan-id 100
```
6. Add the member interfaces to the LAG between the two MC-LAG switches:
 

```
[edit interfaces]
user@switch# set xe-0/0/10 ether-options 802.3ad ae0
user@switch# set xe-0/0/11 ether-options 802.3ad ae0
```
  7. Add the member interfaces to the MC-LAG:
 

```
[edit interfaces]
user@switch# set xe-0/0/20 ether-options 802.3ad ae1
user@switch# set xe-0/0/21 ether-options 802.3ad ae1
```
  8. Configure the interface mode as **trunk** and membership in the FCoE VLAN (**fcoe\_vlan**) for the LAG (**ae0**) and for the MC-LAG (**ae1**):
 

```
[edit interfaces]
user@switch# set interfaces ae0 unit 0 family ethernet-switching interface-mode trunk
vlan members fcoe_vlan
user@switch# set interfaces ae1 unit 0 family ethernet-switching interface-mode trunk
vlan members fcoe_vlan
```
  9. Set the MTU to **2180** for the LAG and MC-LAG interfaces. 2180 bytes is the minimum size required to handle FCoE packets because of the payload and header sizes; you can configure the MTU to a higher number of bytes if desired, but not less than 2180 bytes:
 

```
[edit interfaces]
user@switch# set ae0 mtu 2180
user@switch# set ae1 mtu 2180
```
  10. Set the LAG and MC-LAG interfaces as FCoE trusted ports. Ports that connect to other switches should be trusted and should not perform FIP snooping:
 

```
[edit]
user@switch# set vlans fcoe_vlan forwarding-options fip-security interface ae0 fcoe-trusted
user@switch# set vlans fcoe_vlan forwarding-options fip-security interface ae1 fcoe-trusted
```

### MC-LAG Switches S1 and S2 ETS Hierarchical Scheduling Configuration

#### Step-by-Step Procedure

To configure the forwarding class set (priority group) and priority group scheduling (in a traffic control profile), and apply the ETS hierarchical scheduling for FCoE traffic to interfaces:

1. Configure the forwarding class set (**fcoe-pg**) for the FCoE traffic:
 

```
[edit class-of-service]
user@switch# set forwarding-class-sets fcoe-pg class fcoe
```
2. Define the traffic control profile (**fcoe-tcp**) to use on the FCoE forwarding class set:
 

```
[edit class-of-service]
user@switch# set traffic-control-profiles fcoe-tcp scheduler-map fcoe-map
guaranteed-rate 3g
user@switch# set traffic-control-profiles fcoe-tcp shaping-rate percent 100
```
3. Apply the FCoE forwarding class set and traffic control profile to the LAG and MC-LAG interfaces:
 

```
[edit class-of-service]
```

```

user@switch# set interfaces ae0 forwarding-class-set fcoe-pg output-traffic-control-profile
fcoe-tcp
user@switch# set interfaces ae1 forwarding-class-set fcoe-pg output-traffic-control-profile
fcoe-tcp

```

### MC-LAG Switches S1 and S2 Port Scheduling Configuration

#### Step-by-Step Procedure

To apply port scheduling for FCoE traffic to interfaces:

1. Apply the scheduler map to the egress ports:

```

set class-of-service interfaces ae0 scheduler-map fcoe-map
set class-of-service interfaces ae1 scheduler-map fcoe-map

```

### FCoE Transit Switches TS1 and TS2 Common Configuration (Applies to ETS and Port Scheduling)

#### Step-by-Step Procedure

The CoS configuration on FCoE Transit Switches TS1 and TS2 is similar to the CoS configuration on MC-LAG Switches S1 and S2. However, the port configurations differ, and you must enable FIP snooping on the Switch TS1 and Switch TS2 FCoE access ports.

To configure queue scheduling, PFC, the FCoE VLAN, and LAG interface membership and characteristics to support lossless FCoE transport across the MC-LAG (this example uses the default **fcoe** forwarding class and the default classifier to map incoming FCoE traffic to the FCoE IEEE 802.1p code point 011, so you do not configure them), or both ETS hierarchical scheduling and port scheduling (common configuration):

1. Configure output scheduling for the FCoE queue:

```

[edit class-of-service]
user@switch# set schedulers fcoe-sched priority low transmit-rate 3g
user@switch# set schedulers fcoe-sched shaping-rate percent 100

```

2. Map the FCoE forwarding class to the FCoE scheduler (**fcoe-sched**):

```

[edit class-of-service]
user@switch# set scheduler-maps fcoe-map forwarding-class fcoe scheduler fcoe-sched

```

3. Enable PFC on the FCoE priority by creating a congestion notification profile (**fcoe-cnp**) that applies FCoE to the IEEE 802.1 code point 011:

```

[edit class-of-service]
user@switch# set congestion-notification-profile fcoe-cnp input ieee-802.1 code-point
011 pfc

```

4. Apply the PFC configuration to the LAG interface and to the FCoE access interfaces:

```

[edit class-of-service]
user@switch# set interfaces ae1 congestion-notification-profile fcoe-cnp
user@switch# set class-of-service interfaces xe-0/0/30 congestion-notification-profile
fcoe-cnp
user@switch# set class-of-service interfaces xe-0/0/31 congestion-notification-profile
fcoe-cnp
user@switch# set class-of-service interfaces xe-0/0/32 congestion-notification-profile
fcoe-cnp
user@switch# set class-of-service interfaces xe-0/0/33 congestion-notification-profile
fcoe-cnp

```

5. Configure the VLAN for FCoE traffic (**fcoe\_vlan**):


- ```
[edit vlans]
user@switch# set fcoe_vlan vlan-id 100
```
6. Add the member interfaces to the LAG:


```
[edit interfaces]
user@switch# set xe-0/0/25 ether-options 802.3ad ae1
user@switch# set xe-0/0/26 ether-options 802.3ad ae1
```
 7. On the LAG (**ae1**), configure the interface mode as **trunk** and membership in the FCoE VLAN (**fcoe_vlan**):


```
[edit interfaces]
user@switch# set interfaces ae1 unit 0 family ethernet-switching interface-mode trunk
vlan members fcoe_vlan
```
 8. On the FCoE access interfaces (**xe-0/0/30**, **xe-0/0/31**, **xe-0/0/32**, **xe-0/0/33**), configure the interface mode as **trunk** and membership in the FCoE VLAN (**fcoe_vlan**):


```
[edit interfaces]
user@switch# set interfaces xe-0/0/30 unit 0 family ethernet-switching interface-mode trunk
vlan members fcoe_vlan
user@switch# set interfaces xe-0/0/31 unit 0 family ethernet-switching interface-mode trunk
vlan members fcoe_vlan
user@switch# set interfaces xe-0/0/32 unit 0 family ethernet-switching interface-mode trunk
vlan members fcoe_vlan
user@switch# set interfaces xe-0/0/33 unit 0 family ethernet-switching interface-mode trunk
vlan members fcoe_vlan
```
 9. Set the MTU to **2180** for the LAG and FCoE access interfaces. 2180 bytes is the minimum size required to handle FCoE packets because of the payload and header sizes; you can configure the MTU to a higher number of bytes if desired, but not less than 2180 bytes:


```
[edit interfaces]
user@switch# set ae1 mtu 2180
user@switch# set xe-0/0/30 mtu 2180
user@switch# set xe-0/0/31 mtu 2180
user@switch# set xe-0/0/32 mtu 2180
user@switch# set xe-0/0/33 mtu 2180
```
 10. Set the LAG interface as an FCoE trusted port. Ports that connect to other switches should be trusted and should not perform FIP snooping:


```
[edit]
user@switch# set vlans fcoe_vlan forwarding-options fip-security interface ae1 fcoe-trusted
```
- 

NOTE: Access ports **xe-0/0/30**, **xe-0/0/31**, **xe-0/0/32**, and **xe-0/0/33** are not configured as FCoE trusted ports. The access ports remain in the default state as untrusted ports because they connect directly to FCoE devices and must perform FIP snooping to ensure network security.
11. Enable FIP snooping on the FCoE VLAN to prevent unauthorized FCoE network access (this example uses VN2VN_Port FIP snooping; the example is equally valid if you use VN2VF_Port FIP snooping):


```
[edit]
```



```
user@switch# set vlans fcoe_vlan forwarding-options fip-security examine-vn2vn
beacon-period 90000
```



NOTE: QFX10000 switches do not support FIP snooping and cannot be used as FCoE access transit switches. (QFX10000 switches can be used as FCoE aggregation switches.)

FCoE Transit Switches TS1 and TS2 ETS Hierarchical Scheduling Configuration

Step-by-Step Procedure

To configure the forwarding class set (priority group) and priority group scheduling (in a traffic control profile), and apply the ETS hierarchical scheduling for FCoE traffic to interfaces:

1. Configure the forwarding class set (**fcoe-pg**) for the FCoE traffic:


```
[edit class-of-service]
user@switch# set forwarding-class-sets fcoe-pg class fcoe
```
2. Define the traffic control profile (**fcoe-tcp**) to use on the FCoE forwarding class set:


```
[edit class-of-service]
user@switch# set traffic-control-profiles fcoe-tcp scheduler-map fcoe-map
guaranteed-rate 3g
user@switch# set traffic-control-profiles fcoe-tcp shaping-rate percent 100
```
3. Apply the FCoE forwarding class set and traffic control profile to the LAG interface and to the FCoE access interfaces:


```
[edit class-of-service]
user@switch# set interfaces ae1 forwarding-class-set fcoe-pg output-traffic-control-profile fcoe-tcp
user@switch# set class-of-service interfaces xe-0/0/30 forwarding-class-set fcoe-pg output-traffic-control-profile fcoe-tcp
user@switch# set class-of-service interfaces xe-0/0/31 forwarding-class-set fcoe-pg output-traffic-control-profile fcoe-tcp
user@switch# set class-of-service interfaces xe-0/0/32 forwarding-class-set fcoe-pg output-traffic-control-profile fcoe-tcp
user@switch# set class-of-service interfaces xe-0/0/33 forwarding-class-set fcoe-pg output-traffic-control-profile fcoe-tcp
```

FCoE Transit Switches TS1 and TS2 Port Scheduling Configuration

Step-by-Step Procedure

To apply port scheduling for FCoE traffic to interfaces:

1. Apply the scheduler map to the egress ports:


```
user@switch# set class-of-service interfaces ae1 scheduler-map fcoe-map
user@switch# set class-of-service interfaces xe-0/0/30 scheduler-map fcoe-map
user@switch# set class-of-service interfaces xe-0/0/31 scheduler-map fcoe-map
user@switch# set class-of-service interfaces xe-0/0/32 scheduler-map fcoe-map
user@switch# set class-of-service interfaces xe-0/0/33 scheduler-map fcoe-map
```

Results

Display the results of the CoS configuration on MC-LAG Switch S1 and on MC-LAG Switch S2 (the results on both switches are the same). The results are from the ETS hierarchical scheduling configuration, which shows the more complex configuration. Direct port scheduling results would not show the traffic control profile or forwarding class set portions of the configuration, but would display the name of the scheduler map under each interface (instead of the names of the forwarding class set and output traffic control profile). Other than that, they are the same.

```

user@switch> show configuration class-of-service
traffic-control-profiles {
  fcoe-tcp {
    scheduler-map fcoe-map;
    shaping-rate percent 100;
    guaranteed-rate 3000000000;
  }
}
forwarding-class-sets {
  fcoe-pg {
    class fcoe;
  }
}
congestion-notification-profile {
  fcoe-cnp {
    input {
      ieee-802.1 {
        code-point 011 {
          pfc;
        }
      }
    }
  }
}
interfaces {
  ae0 {
    forwarding-class-set {
      fcoe-pg {
        output-traffic-control-profile fcoe-tcp;
      }
    }
    congestion-notification-profile fcoe-cnp;
  }
  ae1 {
    forwarding-class-set {
      fcoe-pg {
        output-traffic-control-profile fcoe-tcp;
      }
    }
    congestion-notification-profile fcoe-cnp;
  }
}
scheduler-maps {
  fcoe-map {

```

```

        forwarding-class fcoe scheduler fcoe-sched;
    }
}
schedulers {
    fcoe-sched {
        transmit-rate 30000000000;
        shaping-rate percent 100;
        priority low;
    }
}

```



NOTE: The forwarding class and classifier configurations are not shown because the show command does not display default portions of the configuration.

For MC-LAG verification commands, see *Example: Configuring Multichassis Link Aggregation*.

Display the results of the CoS configuration on FCoE Transit Switch TS1 and on FCoE Transit Switch TS2 (the results on both transit switches are the same). The results are from the ETS hierarchical port scheduling configuration, which shows the more complex configuration. Direct port scheduling results would not show the traffic control profile or forwarding class set portions of the configuration, but would display the name of the scheduler map under each interface (instead of the names of the forwarding class set and output traffic control profile). Other than that, they are the same.

```

user@switch> show configuration class-of-service
traffic-control-profiles {
    fcoe-tcp {
        scheduler-map fcoe-map;
        shaping-rate percent 100;
        guaranteed-rate 30000000000;
    }
}
forwarding-class-sets {
    fcoe-pg {
        class fcoe;
    }
}
congestion-notification-profile {
    fcoe-cnp {
        input {
            ieee-802.1 {
                code-point 011 {
                    pfc;
                }
            }
        }
    }
}
}
interfaces {
    xe-0/0/30 {

```

```
forwarding-class-set {
  fcoe-pg {
    output-traffic-control-profile fcoe-tcp;
  }
}
congestion-notification-profile fcoe-cnp;
}
xe-0/0/31 {
  forwarding-class-set {
    fcoe-pg {
      output-traffic-control-profile fcoe-tcp;
    }
  }
  congestion-notification-profile fcoe-cnp;
}
xe-0/0/32 {
  forwarding-class-set {
    fcoe-pg {
      output-traffic-control-profile fcoe-tcp;
    }
  }
  congestion-notification-profile fcoe-cnp;
}
xe-0/0/33 {
  forwarding-class-set {
    fcoe-pg {
      output-traffic-control-profile fcoe-tcp;
    }
  }
  congestion-notification-profile fcoe-cnp;
}
ae1 {
  forwarding-class-set {
    fcoe-pg {
      output-traffic-control-profile fcoe-tcp;
    }
  }
  congestion-notification-profile fcoe-cnp;
}
}
scheduler-maps {
  fcoe-map {
    forwarding-class fcoe scheduler fcoe-sched;
  }
}
schedulers {
  fcoe-sched {
    transmit-rate 3000000000;
    shaping-rate percent 100;
    priority low;
  }
}
```



NOTE: The forwarding class and classifier configurations are not shown because the `show` command does not display default portions of the configuration.

Verification

To verify that the CoS components and FIP snooping have been configured and are operating properly, perform these tasks. Because this example uses the default `fcoe` forwarding class and the default IEEE 802.1p trusted classifier, the verification of those configurations is not shown:

- [Verifying That the Output Queue Schedulers Have Been Created on page 61](#)
- [Verifying That the Priority Group Output Scheduler \(Traffic Control Profile\) Has Been Created \(ETS Configuration Only\) on page 62](#)
- [Verifying That the Forwarding Class Set \(Priority Group\) Has Been Created \(ETS Configuration Only\) on page 62](#)
- [Verifying That Priority-Based Flow Control Has Been Enabled on page 63](#)
- [Verifying That the Interface Class of Service Configuration Has Been Created on page 64](#)
- [Verifying That the Interfaces Are Correctly Configured on page 66](#)
- [Verifying That FIP Snooping Is Enabled on the FCoE VLAN on FCoE Transit Switches TS1 and TS2 Access Interfaces on page 68](#)
- [Verifying That the FIP Snooping Mode Is Correct on FCoE Transit Switches TS1 and TS2 on page 69](#)

Verifying That the Output Queue Schedulers Have Been Created

Purpose Verify that the output queue scheduler for FCoE traffic has the correct bandwidth parameters and priorities, and is mapped to the correct forwarding class (output queue). Queue scheduler verification is the same on each of the four switches.

Action List the scheduler map using the operational mode command `show class-of-service scheduler-map fcoe-map`:

```
user@switch> show class-of-service scheduler-map fcoe-map
Scheduler map: fcoe-map, Index: 9023
```

```
Scheduler: fcoe-sched, Forwarding class: fcoe, Index: 37289
Transmit rate: 3000000000 bps, Rate Limit: none, Buffer size: remainder,
Buffer Limit: none, Priority: low
Excess Priority: unspecified
Shaping rate: 100 percent,
drop-profile-map-set-type: mark
Drop profiles:
  Loss priority  Protocol  Index  Name
  Low           any       1      <default-drop-profile>
  Medium high   any       1      <default-drop-profile>
  High          any       1      <default-drop-profile>
```

Meaning The **show class-of-service scheduler-map fcoe-map** command lists the properties of the scheduler map **fcoe-map**. The command output includes:

- The name of the scheduler map (**fcoe-map**)
- The name of the scheduler (**fcoe-sched**)
- The forwarding classes mapped to the scheduler (**fcoe**)
- The minimum guaranteed queue bandwidth (transmit rate **3000000000 bps**)
- The scheduling priority (**low**)
- The maximum bandwidth in the priority group the queue can consume (shaping rate **100 percent**)
- The drop profile loss priority for each drop profile name. This example does not include drop profiles because you do not apply drop profiles to FCoE traffic.

Verifying That the Priority Group Output Scheduler (Traffic Control Profile) Has Been Created (ETS Configuration Only)

Purpose Verify that the traffic control profile **fcoe-tcp** has been created with the correct bandwidth parameters and scheduler mapping. Priority group scheduler verification is the same on each of the four switches.

Action List the FCoE traffic control profile properties using the operational mode command **show class-of-service traffic-control-profile fcoe-tcp**:

```
user@switch> show class-of-service traffic-control-profile fcoe-tcp
Traffic control profile: fcoe-tcp, Index: 18303
  Shaping rate: 100 percent
  Scheduler map: fcoe-map
  Guaranteed rate: 3000000000
```

Meaning The **show class-of-service traffic-control-profile fcoe-tcp** command lists all of the configured traffic control profiles. For each traffic control profile, the command output includes:

- The name of the traffic control profile (**fcoe-tcp**)
- The maximum port bandwidth the priority group can consume (shaping rate **100 percent**)
- The scheduler map associated with the traffic control profile (**fcoe-map**)
- The minimum guaranteed priority group port bandwidth (guaranteed rate **3000000000** in bps)

Verifying That the Forwarding Class Set (Priority Group) Has Been Created (ETS Configuration Only)

Purpose Verify that the FCoE priority group has been created and that the **fcoe** priority (forwarding class) belongs to the FCoE priority group. Forwarding class set verification is the same on each of the four switches.

Action List the forwarding class sets using the operational mode command **show class-of-service forwarding-class-set fcoe-pg**:

```
user@switch> show class-of-service forwarding-class-set fcoe-pg
Forwarding class set: fcoe-pg, Type: normal-type, Forwarding class set index:
31420
  Forwarding class      Index
  fcoe                  1
```

Meaning The **show class-of-service forwarding-class-set fcoe-pg** command lists all of the forwarding classes (priorities) that belong to the **fcoe-pg** priority group, and the internal index number of the priority group. The command output shows that the forwarding class set **fcoe-pg** includes the forwarding class **fcoe**.

Verifying That Priority-Based Flow Control Has Been Enabled

Purpose Verify that PFC is enabled on the FCoE code point. PFC verification is the same on each of the four switches.

Action List the FCoE congestion notification profile using the operational mode command **show class-of-service congestion-notification fcoe-cnp**:

```
user@switch> show class-of-service congestion-notification fcoe-cnp
Type: Input, Name: fcoe-cnp, Index: 6879
Cable Length: 100 m
  Priority  PFC      MRU
  000      Disabled
  001      Disabled
  010      Disabled
  011      Enabled   2500
  100      Disabled
  101      Disabled
  110      Disabled
  111      Disabled
Type: Output
  Priority  Flow-Control-Queues
  000      0
  001      1
  010      2
  011      3
  100      4
  101      5
  110      6
  111      7
```

Meaning The **show class-of-service congestion-notification fcoe-cnp** command lists all of the IEEE 802.1p code points in the congestion notification profile that have PFC enabled. The command output shows that PFC is enabled on code point **011** (**fcoe** queue) for the **fcoe-cnp** congestion notification profile.

The command also shows the default cable length (100 meters), the default maximum receive unit (2500 bytes), and the default mapping of priorities to output queues because this example does not include configuring these options.

Verifying That the Interface Class of Service Configuration Has Been Created

Purpose Verify that the CoS properties of the interfaces are correct. The verification output on MC-LAG Switches S1 and S2 differs from the output on FCoE Transit Switches TS1 and TS2.



NOTE: The output is from the ETS hierarchical port scheduling configuration to show the more complex configuration. Direct port scheduling results do not show the traffic control profile or forwarding class sets because those elements are configured only for ETS. Instead, the name of the scheduler map is displayed under each interface.

Action List the interface CoS configuration on MC-LAG Switches S1 and S2 using the operational mode command **show configuration class-of-service interfaces**:

```
user@switch> show configuration class-of-service interfaces
ae0 {
    forwarding-class-set {
        fcoe-pg {
            output-traffic-control-profile fcoe-tcp;
        }
    }
    congestion-notification-profile fcoe-cnp;
}

ae1 {
    forwarding-class-set {
        fcoe-pg {
            output-traffic-control-profile fcoe-tcp;
        }
    }
    congestion-notification-profile fcoe-cnp;
}
```

List the interface CoS configuration on FCoE Transit Switches TS1 and TS2 using the operational mode command **show configuration class-of-service interfaces**:

```
user@switch> show configuration class-of-service interfaces
xe-0/0/30 {
    forwarding-class-set {
        fcoe-pg {
            output-traffic-control-profile fcoe-tcp;
        }
    }
    congestion-notification-profile fcoe-cnp;
}
xe-0/0/31 {
    forwarding-class-set {
        fcoe-pg {
```



```

        output-traffic-control-profile fcoe-tcp;
    }
}
congestion-notification-profile fcoe-cnp;
}
xe-0/0/32 {
    forwarding-class-set {
        fcoe-pg {
            output-traffic-control-profile fcoe-tcp;
        }
    }
    congestion-notification-profile fcoe-cnp;
}
xe-0/0/33 {
    forwarding-class-set {
        fcoe-pg {
            output-traffic-control-profile fcoe-tcp;
        }
    }
    congestion-notification-profile fcoe-cnp;
}
ae1 {
    forwarding-class-set {
        fcoe-pg {
            output-traffic-control-profile fcoe-tcp;
        }
    }
    congestion-notification-profile fcoe-cnp;
}

```

Meaning The **show configuration class-of-service interfaces** command lists the class of service configuration for all interfaces. For each interface, the command output includes:

- The name of the interface (for example, **ae0** or **xe-0/0/30**)
- The name of the forwarding class set associated with the interface (**fcoe-pg**)
- The name of the traffic control profile associated with the interface (output traffic control profile, **fcoe-tcp**)
- The name of the congestion notification profile associated with the interface (**fcoe-cnp**)



NOTE: Interfaces that are members of a LAG are not shown individually. The LAG or MC-LAG CoS configuration is applied to all interfaces that are members of the LAG or MC-LAG. For example, the interface CoS configuration output on MC-LAG Switches S1 and S2 shows the LAG CoS configuration but does not show the CoS configuration of the member interfaces separately. The interface CoS configuration output on FCoE Transit Switches TS1 and TS2 shows the LAG CoS configuration but also shows the configuration for interfaces xe-0/0/30, xe-0/0/31, xe-0/0/32, and xe-0/0/33, which are not members of a LAG.

Verifying That the Interfaces Are Correctly Configured

Purpose Verify that the LAG membership, MTU, VLAN membership, and port mode of the interfaces are correct. The verification output on MC-LAG Switches S1 and S2 differs from the output on FCoE Transit Switches T1 and T2.

Action List the interface configuration on MC-LAG Switches S1 and S2 using the operational mode command **show configuration interfaces**:

```
user@switch> show configuration interfaces
xe-0/0/10 {
    ether-options {
        802.3ad ae0;
    }
}
xe-0/0/11 {
    ether-options {
        802.3ad ae0;
    }
}
xe-0/0/20 {
    ether-options {
        802.3ad ae1;
    }
}
xe-0/0/21 {
    ether-options {
        802.3ad ae1;
    }
}
ae0 {
    mtu 2180;
    unit 0 {
        family ethernet-switching {
            interface-mode trunk;
            vlan {
                members fcoe_vlan;
            }
        }
    }
}
ae1 {
    mtu 2180;
    unit 0 {
        family ethernet-switching {
            interface-mode trunk;
            vlan {
                members fcoe_vlan;
            }
        }
    }
}
```

List the interface configuration on FCoE Transit Switches TS1 and TS2 using the operational mode command **show configuration interfaces**:

```
user@switch> show configuration interfaces
```

```
xe-0/0/25 {
  ether-options {
    802.3ad ae1;
  }
}
xe-0/0/26 {
  ether-options {
    802.3ad ae1;
  }
}
xe-0/0/30 {
  mtu 2180;
  unit 0 {
    family ethernet-switching {
      interface-mode trunk;
      vlan {
        members fcoe_vlan;
      }
    }
  }
}
xe-0/0/31 {
  mtu 2180;
  unit 0 {
    family ethernet-switching {
      interface-mode trunk;
      vlan {
        members fcoe_vlan;
      }
    }
  }
}
xe-0/0/32 {
  mtu 2180;
  unit 0 {
    family ethernet-switching {
      interface-mode trunk;
      vlan {
        members fcoe_vlan;
      }
    }
  }
}
xe-0/0/33 {
  mtu 2180;
  unit 0 {
    family ethernet-switching {
      interface-mode trunk;
      vlan {
        members fcoe_vlan;
      }
    }
  }
}
ae1 {
  mtu 2180;
  unit 0 {
    family ethernet-switching {
      interface-mode trunk;
      vlan {
```

```

        members fcoe_vlan;
    }
}
}

```

Meaning The **show configuration interfaces** command lists the configuration of each interface by interface name.

For each interface that is a member of a LAG, the command lists only the name of the LAG to which the interface belongs.

For each LAG interface and for each interface that is not a member of a LAG, the command output includes:

- The MTU (**2180**)
- The unit number of the interface (**0**)
- The interface mode (**trunk** mode both for interfaces that connect two switches and for interfaces that connect to FCoE hosts)
- The name of the VLAN in which the interface is a member (**fcoe_vlan**)

Verifying That FIP Snooping Is Enabled on the FCoE VLAN on FCoE Transit Switches TS1 and TS2 Access Interfaces

Purpose Verify that FIP snooping is enabled on the FCoE VLAN access interfaces. FIP snooping is enabled only on the FCoE access interfaces, so it is enabled only on FCoE Transit Switches TS1 and TS2. FIP snooping is not enabled on MC-LAG Switches S1 and S2 because FIP snooping is done at the Transit Switch TS1 and TS2 FCoE access ports.

Action List the port security configuration on FCoE Transit Switches TS1 and TS2 using the operational mode command **show configuration vlans fcoe_vlan forwarding-options fip-security**:

```

user@switch> show configuration vlans fcoe_vlan forwarding-options fip-security
interface ae1.0 {
    fcoe-trusted;
}
examine-vn2vn {
    beacon-period 90000;
}

```

Meaning The **show configuration vlans fcoe_vlan forwarding-options fip-security** command lists VLAN FIP security information, including whether a port member of the VLAN is trusted. The command output shows that:

- LAG port **ae1.0**, which connects the FCoE transit switch to the MC-LAG switches, is configured as an FCoE trusted interface. FIP snooping is not performed on the member interfaces of the LAG (**xe-0/0/25** and **xe-0/0/26**).
- VN2VN_Port FIP snooping is enabled (**examine-vn2vn**) on the FCoE VLAN and the beacon period is set to 90000 milliseconds. On Transit Switches TS1 and TS2, all

interface members of the FCoE VLAN perform FIP snooping unless the interface is configured as FCoE trusted. On Transit Switches TS1 and TS2, interfaces **xe-0/0/30**, **xe-0/0/31**, **xe-0/0/32**, and **xe-0/0/33** perform FIP snooping because they are not configured as FCoE trusted. The interface members of LAG **ae1** (**xe-0/0/25** and **xe-0/0/26**) do not perform FIP snooping because the LAG is configured as FCoE trusted.

Verifying That the FIP Snooping Mode Is Correct on FCoE Transit Switches TS1 and TS2

Purpose Verify that the FIP snooping mode is correct on the FCoE VLAN. FIP snooping is enabled only on the FCoE access interfaces, so it is enabled only on FCoE Transit Switches TS1 and TS2. FIP snooping is not enabled on MC-LAG Switches S1 and S2 because FIP snooping is done at the Transit Switch TS1 and TS2 FCoE access ports.

Action List the FIP snooping configuration on FCoE Transit Switches TS1 and TS2 using the operational mode command **show fip snooping brief**:

```
user@switch> show fip snooping brief
VLAN: fcoe_vlan,      Mode: VN2VN Snooping
FC-MAP: 0e:fc:00
...
```



NOTE: The output has been truncated to show only the relevant information.

Meaning The **show fip snooping brief** command lists FIP snooping information, including the FIP snooping VLAN and the FIP snooping mode. The command output shows that:

- The VLAN on which FIP snooping is enabled is **fcoe_vlan**
- The FIP snooping mode is VN2VN_Port FIP snooping (**VN2VN Snooping**)

Related Documentation

- *Example: Configuring Multichassis Link Aggregation*
- *Configuring Link Aggregation*
- [Example: Configuring CoS PFC for FCoE Traffic on page 92](#)
- *Example: Configuring CoS Hierarchical Port Scheduling (ETS)*
- *Example: Configuring Queue Schedulers for Port Scheduling*
- *Understanding Multichassis Link Aggregation*
- [Understanding MC-LAGs on an FCoE Transit Switch on page 41](#)

Understanding FCoE and FIP Session High Availability

In FCoE-FC gateway mode, the QFX3500 switch provides high availability to restore the FCoE sessions running on the switch in case the Fibre Channel (FC) process is terminated.

The switch stores FCoE session data in a persistent storage module. If the FC process terminates, the switch restores the existing FCoE sessions on the same interfaces that they were on before the FC process terminated. Data traffic for existing sessions is not affected during session restoration.

For a brief time, the system does not process control traffic because of the FC process restart and session restoration. During this brief time, no new FCoE sessions can be established, and no existing sessions can log out.



NOTE: During the restoration process, if the FC process does not receive an “interface up” notification from a particular interface within a certain time, the switch times out the restore operation and discards the data on that interface. The previously existing FCoE sessions on that interface are not restored, and the ENodes must log in again.



NOTE: An FC process restart and session restoration resets the Fibre Channel statistics.

If the FC process terminates repeatedly, the operating system disables the process until you manually restart it. To restart the FC process manually, issue the **restart fibre-channel** command.

Related Documentation

- *Understanding an FCoE-FC Gateway*
- [Understanding FCoE on page 26](#)
- *Understanding Nonstop Software Upgrade for QFabric Systems*
- *Performing a Nonstop Software Upgrade on the QFabric System*

Troubleshooting Dropped FIP Traffic

Problem **Description:** Fibre Channel over Ethernet (FCoE) Initialization Protocol (FIP) traffic such as FIP VLAN discovery and notification frames is dropped.

Cause The interface on which the FIP traffic is dropped does not have a native VLAN configured. FIP VLAN discovery and notification messages are exchanged as untagged packets on the native VLAN. (After the FCoE session with the Fibre Channel switch is established, FCoE traffic uses the FCoE VLAN.)

Solution Check to ensure that every 10-Gigabit Ethernet interface that connects to an FCoE device includes a native VLAN. Configure a native VLAN on all 10-Gigabit Ethernet interfaces that connect to FCoE devices.



NOTE: Make sure that the native VLAN you are using is the same native VLAN that the FCoE devices use for Ethernet traffic.

The procedure for configuring a native VLAN on an interface is different on switches that use the original CLI than on switches that use the Enhanced Layer 2 Software (ELS) CLI. This topic provides the configuration procedure for each CLI.

Configuring a Native VLAN on Switches Using the Original CLI

To configure a native VLAN on an interface:

1. Set the interface port mode to **tagged-access** if you have not already done so:

```
[edit]
user@switch# set interfaces interface unit unit family ethernet-switching port-mode
tagged-access
```

For example, to set the port mode to **tagged-access** for interface **xe-0/0/6.0**:

```
[edit]
user@switch# set interfaces xe-0/0/6 unit 0 family ethernet-switching port-mode
tagged-access
```

2. Configure the native VLAN if it does not already exist:

```
[edit]
user@switch# set vlans vlan-name vlan-id vlan-id
```

For example, to name the native VLAN **native** and use the VLAN ID 1:

```
[edit]
user@switch# set vlans native vlan-id 1
```

3. Configure the native VLAN on the interface:

```
[edit]
user@switch# set interfaces interface unit unit family ethernet-switching native-vlan-id
vlan-id
```

For example, to configure a native VLAN with the VLAN ID 1 on interface **xe-0/0/6.0**:

```
[edit]
user@switch# set interfaces xe-0/0/6 unit 0 family ethernet-switching native-vlan-id 1
```

Configuring a Native VLAN on Switches Using the ELS CLI

To configure a native VLAN on an interface:

1. Set the interface mode to **trunk** if you have not already done so:

```
[edit]
user@switch# set interfaces interface unit unit family ethernet-switching interface-mode
trunk
```

For example, to set the interface mode to **trunk** for interface **xe-0/0/6.0**:

```
[edit]
user@switch# set interfaces xe-0/0/6 unit 0 family ethernet-switching interface-mode trunk
```

2. Configure the native VLAN if it does not already exist:

```
[edit]
user@switch# set vlans vlan-name vlan-id vlan-id
```

For example, to name the native VLAN **native** and use the VLAN ID 1:

```
[edit]
user@switch# set vlans native vlan-id 1
```

3. Configure the native VLAN on the physical Ethernet interface:

```
[edit]
user@switch# set interfaces interface native-vlan-id vlan-id
```

For example, to configure a native VLAN with the VLAN ID 1 on interface **xe-0/0/6.0**:

```
[edit]
user@switch# set interfaces xe-0/0/6 native-vlan-id 1
```

4. Configure the Ethernet interface as a member of the native VLAN:

```
[edit]
user@switch# set interfaces interface unit unit family ethernet-switching vlan members
vlan-name
```

For example, to configure an Ethernet interface as a member of a native VLAN with the VLAN ID 1 on interface **xe-0/0/6.0**:

```
[edit]
user@switch# set interfaces xe-0/0/6 unit 0 family ethernet-switching vlan members native
```

Related Documentation

- *interfaces*
- *vlans*

- [Configuring VLANs for FCoE Traffic on an FCoE Transit Switch on page 32](#)

Understanding OxID Hash Control for FCoE Traffic Load Balancing on Standalone Switches

The originator exchange identifier (OxID) field is one of several fields that the switch can use in its hash function computation for FCoE traffic load balancing over multiple outgoing links in an Ethernet link aggregation group (LAG) on ports that face an FCoE forwarder (FCF). The originator of an exchange between a pair of Fibre Channel (FC) endpoints (such as an FCoE host and an FC storage device) uses the OxID field as an identifier for that exchange. The originator also uses the OxID field to track the progress of the series of sequences that comprise the exchange.

When FCoE traffic traverses a LAG that faces an FCF, it can take multiple different links between the source and destination endpoints. The idea is to distribute the FCoE traffic across the FCF-facing LAG links, thus balancing the link load. The switch creates a hash value from some of the packet header fields, and uses the hash value to assign each packet to one of the LAG links. The switch always uses five packet header fields to compute the hash value:

- Source ID (SID)
- Destination ID (DID)
- Fabric ID (FID)
- Source Port ID (SPID)
- Source Module ID (SMID)

In addition, the OxID field is included by default in the FCoE load-balancing hash computation. However, if you do not want to use the OxID field in the FCoE load-balancing hash computation, you can remove it from the computation by using the **set forwarding-options hash-key family fcoe oxid disable** command.

Including the OxID field in the load-balancing hash computation allows different exchanges between a pair of Fibre Channel (FC) endpoints (such as an FCoE host and an FC storage device) to take different paths across the network, thus improving the aggregate network throughput.

However, if the paths between different sets of FC endpoints have common links, congestion on one set of FC endpoints can affect the other set of endpoints. Such congestion can happen if the FCoE traffic on the two sets of endpoints uses the same priority (IEEE 802.1p code point). It is common for networks to use priority 3 (IEEE 802.1p code point 011) for FCoE traffic. However, you can assign different IEEE priorities to different lossless FCoE flows as described in *Understanding CoS IEEE 802.1p Priorities for Lossless Traffic Flows* to further separate the traffic flows.

Related Documentation

- [Enabling and Disabling CoS OxID Hash Control on Standalone Switches on page 73](#)

Enabling and Disabling CoS OxID Hash Control on Standalone Switches

The originator exchange identifier (OxID) field is one of several fields that the switch can use in its hash function computation for FCoE traffic load balancing over multiple outgoing links in an Ethernet link aggregation group (LAG) on ports that face an FCoE forwarder (FCF). You can configure whether or not the switch uses the OxID in the hash computation.

Including the OxID field in the load-balancing hash computation allows different exchanges between a pair of Fibre Channel (FC) endpoints (such as an FCoE host and an FC storage device) to take different paths across the network, thus improving the aggregate network throughput.

However, if the paths between different sets of FC endpoints have common links, congestion on one set of FC endpoints can affect the other set of endpoints. Such congestion can happen if the FCoE traffic on the two sets of endpoints uses the same priority (IEEE 802.1p code point). It is common for networks to use priority 3 (IEEE 802.1p code point 011) for FCoE traffic. However, you can assign different IEEE priorities to different lossless FCoE flows as described in *Understanding CoS IEEE 802.1p Priorities for Lossless Traffic Flows* to further separate the traffic flows.

OxID hash control is enabled by default.

- To enable OxID hash control field for FCoE traffic load balancing:

```
[edit forwarding-options hash-key]
user@switch# set family fcoe oxid enable
```

- To disable OxID hash control field for FCoE traffic load balancing:

```
[edit forwarding-options hash-key]
user@switch# set family fcoe oxid disable
```

- Related Documentation**
- [Understanding OxID Hash Control for FCoE Traffic Load Balancing on Standalone Switches on page 72](#)

PART 2

Configuring DCBX and PFC

- [Using DCBX and PFC on page 77](#)

CHAPTER 3

Using DCBX and PFC

- [Understanding DCB Features and Requirements on page 78](#)
- [Understanding CoS Flow Control \(Ethernet PAUSE and PFC\) on page 81](#)
- [Example: Configuring CoS PFC for FCoE Traffic on page 92](#)
- [Understanding DCBX on page 101](#)
- [Configuring the DCBX Mode on page 111](#)
- [Configuring DCBX Autonegotiation on page 112](#)
- [Disabling the ETS Recommendation TLV on page 115](#)
- [Understanding DCBX Application Protocol TLV Exchange on page 116](#)
- [Defining an Application for DCBX Application Protocol TLV Exchange on page 120](#)
- [Configuring an Application Map for DCBX Application Protocol TLV Exchange on page 121](#)
- [Applying an Application Map to an Interface for DCBX Application Protocol TLV Exchange on page 122](#)
- [Example: Configuring DCBX Application Protocol TLV Exchange on page 123](#)

Understanding DCB Features and Requirements

Data center bridging (DCB) is a set of enhancements to the IEEE 802.1 bridge specifications. DCB modifies and extends Ethernet behavior to support I/O convergence in the data center. I/O convergence includes but is not limited to the transport of Ethernet LAN traffic and Fibre Channel (FC) storage area network (SAN) traffic on the same physical Ethernet network infrastructure.

A converged architecture saves cost by reducing the number of networks and switches required to support both types of traffic, reducing the number of interfaces required, reducing cable complexity, and reducing administration activities.

The Juniper Networks QFX Series supports the DCB features required to transport converged Ethernet and FC traffic while providing the class-of-service (CoS) and other characteristics FC requires for transmitting storage traffic. To accommodate FC traffic, DCB specifications provide:

- A flow control mechanism called priority-based flow control (PFC, described in IEEE 802.1Qbb) to help provide lossless transport.
- A discovery and exchange protocol for conveying configuration and capabilities among neighbors to ensure consistent configuration across the network, called Data Center Bridging Capability Exchange protocol (DCBX), which is an extension of Link Layer Data Protocol (LLDP, described in IEEE 802.1AB).
- A bandwidth management mechanism called enhanced transmission selection (ETS, described in IEEE 802.1Qaz).
- A congestion management mechanism called quantized congestion notification (QCN, described in IEEE 802.1Qau).

The switch supports the PFC, DCBX, and ETS standards but does not support QCN. The switch also provides the high-bandwidth interfaces (10-Gbps minimum) required to support DCB and converged traffic.



NOTE: The Juniper Networks QFX10000 Series does not support enhanced transmission selection (ETS) hierarchical scheduling in this release. Use port scheduling to manage bandwidth on QFX10000 switches.

This topic describes the DCB standards and requirements the switch supports:

- [Lossless Transport on page 78](#)
- [ETS on page 79](#)
- [DCBX on page 80](#)

Lossless Transport

FC traffic requires lossless transport (defined as no frames dropped because of congestion). Standard Ethernet does not support lossless transport, but the DCB

extensions to Ethernet along with proper buffer management enable an Ethernet network to provide the level of class of service (CoS) necessary to transport FC frames encapsulated in Ethernet over an Ethernet network.

This section describes these factors in creating lossless transport over Ethernet:

- [PFC on page 79](#)
- [Buffer Management on page 79](#)
- [Physical Interfaces on page 79](#)

PFC

PFC is a link-level flow control mechanism similar to Ethernet PAUSE (described in IEEE 802.3x). Ethernet PAUSE stops all traffic on a link for a period of time. PFC enables you to divide traffic on a link into eight priorities and stop the traffic of a selected priority without stopping the traffic assigned to other priorities on the link.

Pausing the traffic of a selected priority enables you to provide lossless transport for traffic assigned that priority and at the same time use standard lossy Ethernet transport for the rest of the link traffic.

Buffer Management

Buffer management is critical to the proper functioning of PFC, because if buffers are allowed to overflow, frames are dropped and transport is not lossless.

For each lossless flow priority, the switch requires sufficient buffer space to:

- Store frames sent during the time it takes to send the PFC pause frame across the cable between devices.
- Store the frames that are already on the wire when the sender receives the PFC pause frame.

The propagation delay due to cable length and speed, as well as processing speed, determines the amount of buffer space needed to prevent frame loss due to congestion.

The switch automatically sets the threshold for sending PFC pause frames to accommodate delay from cables as long as 300 meters (984 feet) and to accommodate large frames that might be on the wire when the switch sends the pause frame. This ensures that the switch sends pause frames early enough to allow the sender to stop transmitting before the receive buffers on the switch overflow.

Physical Interfaces

The switch supports 10-Gbps, full-duplex interfaces. The switch enables DCB capability only on 10-Gbps (or faster) Ethernet interfaces.

ETS

PFC divides traffic into up to eight separate streams (priorities, configured on the switch as forwarding classes) on a physical link. ETS enables you to manage the link bandwidth by:

- Grouping the priorities into priority groups (configured on the switch as forwarding class sets).
- Specifying the bandwidth available to each of the priority groups as a percentage of the total available link bandwidth.
- Allocating the bandwidth to the individual priorities in the priority group.

The available link bandwidth is the bandwidth remaining after servicing strict priority flows.

Managing link bandwidth with ETS provides several advantages:

- There is uniform management of all types of traffic on the link, both congestion-managed traffic and standard Ethernet traffic.
- When a priority group does not use all of its allocated bandwidth, other priority groups on the link can use that bandwidth as needed.

When a priority in a priority group does not use all of its allocated bandwidth, other priorities in the group can use that bandwidth.

The result is better bandwidth utilization, because priorities that consist of bursty traffic can share bandwidth during periods of low traffic transmission instead of consuming their entire bandwidth allocation when traffic loads are light.

- You can assign traffic types with different service needs to different priorities so that each traffic type receives appropriate treatment.
- Strict priority traffic retains its allocated bandwidth.

DCBX

DCB devices use DCBX to exchange configuration information with directly connected peers (switches and endpoints such as servers). DCBX is an extension of LLDP. If you disable LLDP on an interface, that interface cannot run DCBX. If you attempt to enable DCBX on an interface on which LLDP is disabled, the configuration commit fails.

DCBX can:

- Discover the DCB capabilities of peers.
- Detect DCB feature misconfiguration or mismatches between peers.
- Configure DCB features on peers.

You can configure DCBX operation for PFC, the FCoE application, and ETS. DCBX is enabled or disabled on a per-interface basis.

Related Documentation

- *Overview of Fibre Channel*
- [Understanding FCoE on page 26](#)
- *Understanding CoS Hierarchical Port Scheduling (ETS)*
- [Understanding CoS Flow Control \(Ethernet PAUSE and PFC\) on page 81](#)

- [Understanding DCBX on page 101](#)
- [Understanding Fibre Channel Terminology](#)
- [Example: Configuring CoS PFC for FCoE Traffic on page 92](#)

Understanding CoS Flow Control (Ethernet PAUSE and PFC)

Flow control supports lossless transmission by regulating traffic flows to avoid dropping frames during periods of congestion. Flow control stops and resumes the transmission of network traffic between two connected peer nodes on a full-duplex Ethernet physical link. Controlling the flow by pausing and restarting it prevents buffers on the nodes from overflowing and dropping frames. You configure flow control on a per-interface basis.

Two methods of peer-to-peer flow control are supported:

- IEEE 802.3X Ethernet PAUSE



NOTE: QFX10000 switches do not support Ethernet PAUSE. Information about Ethernet PAUSE does not apply to QFX10000 switches.

OCX Series switches support symmetric Ethernet PAUSE flow control on Layer 3 tagged interfaces. OCX Series switches do not support asymmetric Ethernet PAUSE flow control. Information about asymmetric flow control does not apply to OCX Series switches.

- IEEE 802.1Qbb priority-based flow control (PFC)



NOTE: OCX Series switches do not support PFC or lossless Layer 2 transport. Information about PFC, lossless transport, and congestion notification profiles do not apply to OCX Series switches.



Video: [Why Use PFC in a Data Center Network?](#)

- [General Information about Ethernet PAUSE and PFC and When to Use Them on page 81](#)
- [Ethernet PAUSE on page 82](#)
- [PFC on page 87](#)
- [Lossless Transport Support Summary on page 90](#)

General Information about Ethernet PAUSE and PFC and When to Use Them

Ethernet PAUSE and PFC are link-level flow control mechanisms.



NOTE: For end-to-end congestion control for best-effort traffic, see *Understanding CoS Explicit Congestion Notification*.

Ethernet PAUSE pauses transmission of all traffic on a physical Ethernet link.

PFC decouples the pause function from the physical Ethernet link and enables you to divide traffic on one link into eight priorities. You can think of the eight priorities as eight “lanes” of traffic that are mapped to forwarding classes and output queues. Each priority maps to a 3-bit IEEE 802.1p CoS code point value in the VLAN header. You can enable PFC on one or more priorities (IEEE 802.1p code points) on a link. When PFC-enabled traffic is paused on a link, traffic that is not PFC-enabled continues to flow (or is dropped if congestion is severe enough).

Use Ethernet PAUSE when you want to prevent packet loss on all of the traffic on a link. Use PFC to prevent traffic loss only on a specified type of traffic that require lossless treatment, for example, Fibre Channel over Ethernet (FCoE) traffic.



NOTE: Depending on the amount of traffic on a link or assigned to a priority, pausing traffic can cause ingress port congestion and spread congestion through the network.

Ethernet PAUSE and PFC are mutually exclusive configurations on an interface. Attempting to configure both Ethernet PAUSE and PFC on a link causes a commit error.

By default, all forms of flow control are disabled. You must explicitly enable flow control on interfaces to pause traffic.

Ethernet PAUSE

Ethernet PAUSE is a congestion relief feature that works by providing link-level flow control for all traffic on a full-duplex Ethernet link. Ethernet PAUSE works in both directions on the link. In one direction, an interface generates and sends Ethernet PAUSE messages to stop the connected peer from sending more traffic. In the other direction, the interface responds to Ethernet PAUSE messages it receives from the connected peer to stop sending traffic.



NOTE: QFX10000 switches do not support Ethernet PAUSE. Information about Ethernet PAUSE does not apply to QFX10000 switches.

OCX Series switches support symmetric Ethernet PAUSE flow control on Layer 3 tagged interfaces. OCX Series switches do not support asymmetric Ethernet PAUSE flow control. Information about asymmetric flow control does not apply to OCX Series switches.

Ethernet PAUSE also works on aggregated Ethernet interfaces. For example, if the connected peer interfaces are called Node A and Node B:

- When the receive buffers on interface Node A reach a certain level of fullness, the interface generates and sends an Ethernet PAUSE message to the connected peer (interface Node B) to tell the peer to stop sending frames. The Node B buffers store frames until the time period specified in the Ethernet PAUSE frame elapses; then Node B resumes sending frames to Node A.
- When interface Node A receives an Ethernet PAUSE message from interface Node B, interface Node A stops transmitting frames until the time period specified in the Ethernet PAUSE frame elapses; then Node A resumes transmission. (The Node A transmit buffers store frames until Node A resumes sending frames to Node B.)

In this scenario, if Node B sends an Ethernet PAUSE frame with a time value of 0 to Node A, the 0 time value indicates to Node A that it can resume transmission. This happens when the Node B buffer empties to below a certain threshold and the buffer can once again accept traffic.

Symmetric flow control means an interface has the same Ethernet PAUSE configuration in both directions. The Ethernet PAUSE generation and Ethernet PAUSE response functions are both configured as enabled, or they are both disabled. You configure symmetric flow control by including the **flow-control** statement at the **[edit interfaces interface-name ether-options]** hierarchy level.

Asymmetric flow control allows you to configure the Ethernet PAUSE functionality in each direction independently on an interface. The configuration for generating Ethernet PAUSE messages and for responding to Ethernet PAUSE messages does not have to be the same. It can be enabled in both directions, disabled in both directions, or enabled in one direction and disabled in the other direction. You configure asymmetric flow control by including the **configured-flow-control** statement at the **[edit interfaces interface-name ether-options]** hierarchy level.

On any particular interface, symmetric and asymmetric flow control are mutually exclusive. Asymmetric flow control overrides and disables symmetric flow control. (If PFC is configured on an interface, you cannot commit an Ethernet PAUSE configuration on the interface. Attempting to commit an Ethernet PAUSE configuration on an interface with PFC enabled on one or more queues results in a commit error. To commit the PAUSE configuration, you must first delete the PFC configuration.) Both symmetric and asymmetric flow control are supported.

- [Symmetric Flow Control on page 83](#)
- [Asymmetric Flow Control on page 84](#)

Symmetric Flow Control

Symmetric flow control configures both the receive and transmit buffers in the same state. The interface can both send Ethernet PAUSE messages and respond to them (flow control is enabled), or the interface cannot send Ethernet PAUSE messages or respond to them (flow control is disabled).

When you enable symmetric flow control on an interface, the Ethernet PAUSE behavior depends on the configuration of the connected peer. With symmetric flow control enabled, the interface can perform any Ethernet PAUSE functions that the connected peer can

perform. (When symmetric flow control is disabled, the interface does not send or respond to Ethernet PAUSE messages.)

Asymmetric Flow Control

Asymmetric flow control enables you to specify independently whether or not the interface receive buffer generates and sends Ethernet PAUSE messages to stop the connected peer from transmitting traffic, and whether or not the interface transmit buffer responds to Ethernet PAUSE messages it receives from the connected peer and stops transmitting traffic. The receive buffer configuration determines if the interface transmits Ethernet PAUSE messages, and the transmit buffer configuration determines if the interface receives and responds to Ethernet PAUSE messages:

- Receive buffers on—Enable Ethernet PAUSE transmission (generate and send Ethernet PAUSE frames)
- Transmit buffers on—Enable Ethernet PAUSE reception (respond to received Ethernet PAUSE frames)

You must explicitly set the flow control for both the receive buffer and the transmit buffer (**on** or **off**) to configure asymmetric Ethernet PAUSE. [Table 4 on page 84](#) describes the configured flow control state when you set the receive (Rx) and transmit (Tx) buffers on an interface:

Table 4: Asymmetric Ethernet PAUSE Flow Control Configuration

Receive (Rx) Buffer	Transmit (Tx) Buffer	Configured Flow Control State
On	Off	Interface generates and sends Ethernet PAUSE messages. Interface does not respond to Ethernet PAUSE messages (interface continues to transmit even if peer requests that the interface stop sending traffic).
Off	On	Interface responds to Ethernet PAUSE messages received from the connected peer, but does not generate or send Ethernet PAUSE messages. (The interface does not request that the connected peer stop sending traffic.)
On	On	Same functionality as symmetric Ethernet PAUSE. Interface generates and sends Ethernet PAUSE messages and responds to received Ethernet PAUSE messages.
Off	Off	Ethernet PAUSE flow control is disabled.

The configured flow control is the Ethernet PAUSE state configured on the interface.

On 1-Gigabit Ethernet interfaces, autonegotiation of Ethernet PAUSE with the connected peer is supported. (Autonegotiation on 10-Gigabit Ethernet interfaces is not supported.) Autonegotiation enables the interface to exchange state advertisements with the connected peer so that the two devices can agree on the Ethernet PAUSE configuration. Each interface advertises its flow control state to the connected peer using a combination of the Ethernet PAUSE and ASM_DIR bits, as described in [Table 5 on page 85](#):

Table 5: Flow Control State Advertised to the Connected Peer (Autonegotiation)

Rx Buffer State	Tx Buffer State	PAUSE Bit	ASM_DIR Bit	Description
Off	Off	0	0	The interface advertises no Ethernet PAUSE capability. This is equivalent to disabling flow control on an interface.
On	On	1	0	The interface advertises symmetric flow control (both the transmission of Ethernet PAUSE messages and the ability to receive and respond to Ethernet PAUSE messages).
On	Off	0	1	The interface advertises asymmetric flow control (the transmission of Ethernet PAUSE messages, but not the ability to receive and respond to Ethernet PAUSE messages).
Off	On	1	1	The interface advertises both symmetric and asymmetric flow control. Although the interface does not generate and send Ethernet PAUSE requests to the peer, the interface supports both symmetric and asymmetric Ethernet PAUSE configuration on the peer because the peer is not affected if the peer does not receive Ethernet PAUSE requests. (If the interface responds to the peer's Ethernet PAUSE requests, that is sufficient to support either symmetric or asymmetric flow control on the peer.)

The flow control configuration on each switch interface interacts with the flow control configuration of the connected peer. Each peer advertises its state to the other peer. The interaction of the flow control configuration of the peers determines the flow control

behavior (resolution) between them, as shown in [Table 6 on page 86](#). The first four columns show the Ethernet PAUSE configuration on the local QFX Series or EX4600 switch and on the connected peer (also known as the *link partner*). The last two columns show the Ethernet PAUSE resolution that results from the local and peer configurations on each interface. This illustrates how the Ethernet PAUSE configuration of each interface affects the Ethernet PAUSE behavior on the other interface.



NOTE: In the Resolution columns of the table, disabling Ethernet PAUSE transmit means that the interface receive buffers do not generate and send Ethernet PAUSE messages to the peer. Disabling Ethernet PAUSE receive means that the interface transmit buffers do not respond to Ethernet PAUSE messages received from the peer.

Table 6: Asymmetric Ethernet PAUSE Behavior on Local and Peer Interfaces

Local Interface (QFX Series or EX4600 Switch)		Peer Interface		Local Resolution	Peer Resolution
PAUSE Bit	ASM_DIR Bit	PAUSE Bit	ASM_DIR Bit		
0	0	Don't care	Don't care	Disable Ethernet PAUSE transmit and receive	Disable Ethernet PAUSE transmit and receive
0	1	0	Don't care	Disable Ethernet PAUSE transmit and receive	Disable Ethernet PAUSE transmit and receive
0	1	1	0	Disable Ethernet PAUSE transmit and receive	Disable Ethernet PAUSE transmit and receive
0	1	1	1	Enable Ethernet PAUSE transmit and disable Ethernet PAUSE receive	Disable Ethernet PAUSE transmit and enable Ethernet PAUSE receive
1	0	0	Don't care	Disable Ethernet PAUSE transmit and receive	Disable Ethernet PAUSE transmit and receive
1	0	1	Don't care	Enable Ethernet PAUSE transmit and receive	Enable Ethernet PAUSE transmit and receive
1	1	0	0	Disable Ethernet PAUSE transmit and receive	Disable Ethernet PAUSE transmit and receive
1	1	0	1	Enable Ethernet PAUSE receive and disable Ethernet PAUSE transmit	Enable Ethernet PAUSE transmit and disable Ethernet PAUSE receive
1	1	Don't care	Don't care	Enable Ethernet PAUSE transmit and receive	Enable Ethernet PAUSE transmit and receive



NOTE: For your convenience, [Table 6 on page 86](#) replicates Table 28B-3 of Section 2 of the IEEE 802.X specification.

PFC

PFC is a lossless transport and congestion relief feature that works by providing granular link-level flow control for each IEEE 802.1p code point (priority) on a full-duplex Ethernet link. When the receive buffer on a switch interface fills to a threshold, the switch transmits a pause frame to the sender (the connected peer) to temporarily stop the sender from transmitting more frames. The buffer threshold must be low enough so that the sender has time to stop transmitting frames and the receiver can accept the frames already on the wire before the buffer overflows. The switch automatically sets queue buffer thresholds to prevent frame loss.

When congestion forces one priority on a link to pause, all of the other priorities on the link continue to send frames. Only frames of the paused priority are not transmitted. When the receive buffer empties below another threshold, the switch sends a message that starts the flow again.

You configure PFC using a congestion notification profile (CNP). A CNP has two parts:

- **Input**—Specify the code point (or code points) on which to enable PFC, and optionally specify the maximum receive unit (MRU) and the cable length between the interface and the connected peer interface.
- **Output**—Specify the output queue or output queues that respond to pause messages from the connected peer.

You apply a PFC configuration by configuring a CNP on one or more interfaces. Each interface that uses a particular CNP is enabled to pause traffic identified by the priorities (code points) specified in that CNP. You can configure one CNP on an interface, and you can configure different CNPs on different interfaces. When you configure a CNP on an interface, ingress traffic that is mapped to a priority that the CNP enables for PFC is paused whenever the queue buffer fills to the pause threshold. (The pause threshold is not user-configurable.)

Configure PFC for a priority end to end along the entire data path to create a lossless lane of traffic on the network. You can selectively pause the traffic in any queue without pausing the traffic for other queues on the same link. You can create lossless lanes for traffic such as FCoE, LAN backup, or management, while using standard frame-drop congestion management for IP traffic on the same link.

Potential consequences of flow control are:

- Ingress port congestion (configuring too many lossless flows can cause ingress port congestion)
- A paused priority that causes upstream devices to pause the same priority, thus spreading congestion back through the network

By definition, PFC supports symmetric pause only (as opposed to Ethernet PAUSE, which supports symmetric and asymmetric pause). With symmetric pause, a device can:

- Transmit pause frames to pause incoming traffic. (You configure this using the input stanza of a congestion notification profile.)
- Receive pause frames and stop sending traffic to a device whose buffer is too full to accept more frames. (You configure this using the output stanza of a congestion notification profile.)

Receiving a PFC frame from a connected peer pauses traffic on egress queues based on the IEEE 802.1p priorities that the PFC pause frame identifies. The priorities are 0 through 7. By default, the priorities map to queue numbers 0 through 7, respectively, and to specific forwarding classes, as shown in [Table 7 on page 88](#):

Table 7: Default PFC Priority to Queue and Forwarding Class Mapping

IEEE 802.1p Priority (Code Point)	Queue	Forwarding Class
0 (000)	0	best-effort
1 (001)	1	best-effort
2 (010)	2	best-effort
3 (011)	3	fcoe
4 (100)	4	no-loss
5 (101)	5	best-effort
6 (110)	6	network-control
7 (111)	7	network-control

For example, a received PFC pause frame that pauses priority 3 pauses output queue 3. If you do not want to use the default configuration, you can configure customized mapping of priorities to queues and forwarding classes.



NOTE: By convention, deployments with converged server access typically use IEEE 802.1p priority 3 for FCoE traffic. The default configuration sets the fcoe forwarding class as a lossless forwarding class that is mapped to queue 3. The default classifier maps incoming priority 3 traffic to the fcoe forwarding class. *However, you must apply PFC to the entire FCoE data path to configure the end-to-end lossless behavior that FCoE traffic requires.*

If your network uses priority 3 for FCoE traffic, we recommend that you use the default configuration. If your network uses a priority other than 3 for FCoE traffic, you can configure lossless FCoE transport on any IEEE 802.1p priority as described in *Understanding CoS IEEE 802.1p Priorities for Lossless Traffic Flows* and *Understanding CoS IEEE 802.1p Priority Remapping on an FCoE-FC Gateway*.

To enable PFC on a priority:

1. Specify the IEEE 802.1p code point to pause in the input stanza of a CNP.
2. If you are not using the default lossless forwarding classes, specify the IEEE 802.1p code point to pause and the corresponding output queue in the output stanza of the CNP.
3. Apply the CNP to the ingress interfaces on which you want to pause the traffic.
4. If you are not using the default lossless forwarding classes, apply the CNP to the ingress interfaces on which you want to pause the traffic.



CAUTION: Any change to the PFC configuration on a port temporarily blocks the entire port (not just the priorities affected by the PFC change) so that the port can implement the change, then unblocks the port. Blocking the port stops ingress and egress traffic, and causes packet loss on all queues on the port until the port is unblocked.

A change to the PFC configuration means any change to a CNP, including changing the input portion of the CNP (enabling or disabling PFC on a priority, or changing the MRU or cable-length values) or changing the output portion of the CNP that enables or disables output flow control on a queue. A PFC configuration change only affects ports that use the changed CNP.

The following actions change the PFC configuration:

- Deleting or disabling a PFC configuration (input or output) in a CNP that is in use on one or more interfaces. For example:
 1. An existing CNP with an input stanza that enables PFC on priorities 3, 5, and 6 is configured on interfaces xe-0/0/20 and xe-0/0/21.
 2. We disable the PFC configuration for priority 6 in the input CNP, and then commit the configuration.

3. The PFC configuration change causes all traffic on interfaces xe-0/0/20 and xe-0/0/21 to stop until the PFC change has been implemented. When the PFC change has been implemented, traffic resumes.

- Configuring a CNP on an interface. (This changes the PFC state by enabling PFC on one or more priorities.)
- Deleting a CNP from an interface. (This changes the PFC state by disabling PFC on one or more priorities.)

When you associate the CNP with an interface, the interface uses PFC to send pause requests when the output queue buffer for the lossless traffic fills to the pause threshold.

On switches that use different classifiers for unicast and multdestination traffic, you can map a unicast queue (queue 0 through 7) and a multdestination queue (queue 8, 9, 10, or 11) to the same IEEE 802.1p code point (priority) so that both unicast and multicast traffic use that priority. However, do not map multdestination traffic to lossless output queues. Starting with Junos OS Release 12.3, you can map one priority to multiple output queues.



NOTE: You can attach a maximum of one CNP to an interface, but you can create an unlimited number of CNPs that explicitly configure only the input stanza and use the default output stanza.

The output stanza of the CNP maps to a profile that interfaces use to respond to pause messages received from the connected peer. On standalone switches, you can create two CNPs with an explicitly configured output stanza.

When a switch is a Node device in a QFabric system, you can create one CNP with an explicitly configured output stanza. (One fewer profile is available on QFabric systems because the system needs a default profile for fabric interfaces, which are not used as fabric interfaces when the switches are not part of a QFabric system. *Understanding CoS IEEE 802.1p Priorities for Lossless Traffic Flows* describes configuring output flow control.

Lossless Transport Support Summary

The switch supports up to six lossless forwarding classes. For lossless transport, you must enable PFC on the IEEE 802.1p priorities (code points) mapped to lossless forwarding classes.



CAUTION: Any change to the PFC configuration on a port temporarily blocks the entire port (not just the priorities affected by the PFC change) so that the port can implement the change, then unblocks the port. Blocking the port stops ingress and egress traffic, and causes packet loss on all queues on the port until the port is unblocked.

The following limitation applies to support lossless transport on QFabric systems only:

- The internal fiber cable length from the QFabric system Node device to the QFabric system Interconnect device cannot exceed 150 meters.

The default CoS configuration provides two lossless forwarding classes, *fcoe* and *no-loss*. If you explicitly configure lossless forwarding classes, you must include the **no-loss** packet drop attribute to enable lossless behavior, or the traffic is not lossless. For both default and explicit lossless forwarding class configuration, you must configure CNP input stanzas to enable PFC on the priority of the lossless traffic and apply the CNPs to ingress interfaces.



NOTE: The information in this note applies only to systems that do not run the ELS CLI.

Junos OS Release 12.2 introduced changes to the way the switch handles lossless forwarding classes (including the default *fcoe* and *no-loss* forwarding classes).

In Junos OS Release 12.1, either explicitly configuring the *fcoe* and *no-loss* forwarding classes or using the default configuration for these forwarding classes resulted in the same lossless behavior for traffic mapped to those forwarding classes.

However, in Junos OS Release 12.2, if you explicitly configure the *fcoe* or the *no-loss* forwarding class, that forwarding class is no longer treated as a lossless forwarding class. Traffic mapped to these forwarding classes is treated as lossy (best-effort) traffic. This is true even if the explicit configuration is exactly the same as the default configuration.

If your CoS configuration from Junos OS Release 12.1 or earlier includes the explicit configuration of the *fcoe* or the *no-loss* forwarding class, then when you upgrade to Junos OS Release 12.2, those forwarding classes are not lossless. To preserve the lossless treatment of these forwarding classes, delete the the explicit *fcoe* and *no-loss* forwarding class configuration before you upgrade to Junos OS Release 12.2.

See *Overview of CoS Changes Introduced in Junos OS Release 12.2* for detailed information about this change and how to delete an existing lossless configuration.

In Junos OS Release 12.3, the default behavior of the *fcoe* and *no-loss* forwarding classes is the same as in Junos OS Release 12.2. However, in Junos OS Release 12.3, you can configure up to six lossless forwarding classes. All explicitly configured lossless forwarding classes must include the new *no-loss* packet drop attribute or the forwarding class is lossy.

Understanding CoS IEEE 802.1p Priorities for Lossless Traffic Flows provides detailed information about the explicit configuration of lossless priorities and about the default configuration of lossless priorities, including the input and output stanzas of the CNP.



NOTE: PFC and Ethernet PAUSE are used only on Ethernet interfaces. Fabric (fte) ports on QFabric systems (Node device fabric ports and Interconnect device fabric ports) use link-layer flow control (LLFC) to ensure the appropriate treatment of lossless traffic.

**Related
Documentation**

- [Understanding DCB Features and Requirements on page 78](#)
- [Understanding CoS Explicit Congestion Notification](#)
- [Configuring CoS PFC \(Congestion Notification Profiles\)](#)
- [Example: Configuring CoS PFC for FCoE Traffic on page 92](#)

Example: Configuring CoS PFC for FCoE Traffic

Priority-based flow control (PFC, described in IEEE 802.1Qbb) is a link-level flow control mechanism that you apply at ingress interfaces. PFC enables you to divide traffic on one physical link into eight priorities. You can think of the eight priorities as eight “lanes” of traffic that correspond to queues (forwarding classes). Each priority is mapped to a 3-bit IEEE 802.1p CoS value in the VLAN header.

You can selectively apply PFC to the traffic in any queue without pausing the traffic in other queues on the same link. You must apply PFC to FCoE traffic to ensure lossless transport.

This example describes how to configure PFC for FCoE traffic:

- [Requirements on page 92](#)
- [Overview on page 92](#)
- [Configuration on page 95](#)
- [Verification on page 99](#)

Requirements

This example uses the following hardware and software components:

- One switch
- Junos OS Release 11.1 or later for the QFX Series

Overview

FCoE traffic requires PFC to ensure lossless packet transport. This example shows you how to configure PFC on FCoE traffic, use the default FCoE forwarding-class-to-queue mapping and:

- Configure a classifier that associates the FCoE forwarding class with FCoE traffic, which is identified by IEEE 802.1p code point 011 (priority 3).
- Configure a congestion notification profile to apply PFC to the FCoE traffic.

- Apply the classifier and the PFC configuration to ingress interfaces.



NOTE: Configuring or changing PFC on an interface blocks the entire port until the PFC change is completed. After a PFC change is completed, the port is unblocked and traffic resumes. Blocking the port stops ingress and egress traffic, and causes packet loss on all queues on the port until the port is unblocked.

- Configure the CoS bandwidth scheduling for the FCoE forwarding class output queue.
- On switches that support enhanced transmission selection (ETS) hierarchical port scheduling, create a forwarding class set (priority group) that includes the FCoE forwarding class; this is required to configure enhanced transmission selection (ETS) and support data center bridging (DCB).
- For ETS, configure the bandwidth scheduling for the FCoE priority group.
- Apply the configuration to ingress and egress interfaces. How this is done differs depending on whether you use ETS or direct port scheduling for the CoS configuration.

For direct port scheduling, you apply a scheduler map directly to the interface. A scheduler map maps schedulers to forwarding classes, and applies the CoS properties of the scheduler to the output queue mapped to the forwarding class.

For ETS hierarchical port scheduling, you apply the scheduler map to a traffic control profile, and then apply the traffic control profile to the interface. The scheduler map maps CoS properties to forwarding classes (and their associated output queues) just as it does for direct port scheduling. The traffic control profile maps CoS properties to the priority group (a group of forwarding classes defined in a forwarding class set) that contains the forwarding class, creating a CoS hierarchy that allocates port bandwidth to a group of forwarding classes (priority group), and then allocates the priority group bandwidth to the individual forwarding classes (see *Understanding CoS Hierarchical Port Scheduling (ETS)*).

Each interface in this example acts as both an ingress interface and an egress interface, so the classifier, congestion notification profile, and scheduling are applied to all of the interfaces.

Topology

Table 8 on page 93 shows the configuration components for this example.

Table 8: Components of the PFC for FCoE Traffic Configuration Topology

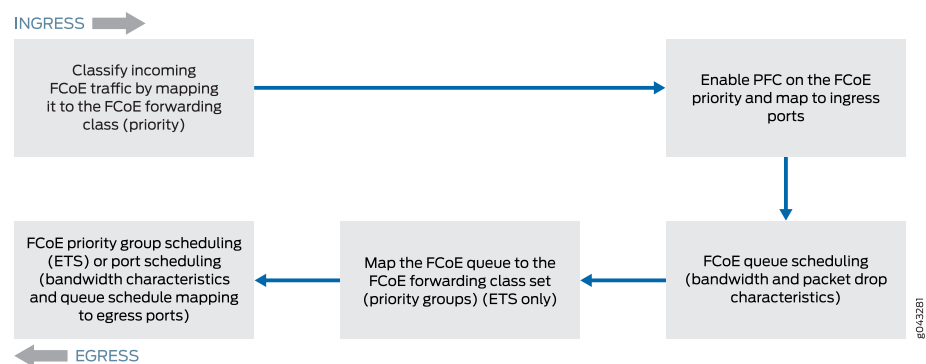
Component	Settings
Hardware	One switch
Behavior aggregate classifier (maps the FCoE forwarding class to incoming packets by IEEE 802.1 code point)	Code point 011 to forwarding class fcoe and loss priority low Ingress interfaces: xe-0/0/31, xe-0/0/32, xe-0/0/33, xe-0/0/34

Table 8: Components of the PFC for FCoE Traffic Configuration Topology (*continued*)

Component	Settings
PFC congestion notification profile	fcoe-cnp: Code point 011 Ingress interfaces: xe-0/0/31, xe-0/0/32, xe-0/0/33, xe-0/0/34
FCoE queue scheduler	fcoe-sched: Minimum bandwidth 3g Maximum bandwidth 100% Priority low
Forwarding class-to-scheduler mapping	Scheduler map fcoe-map: Forwarding class fcoe Scheduler fcoe-sched On switches that support direct port scheduling, if you use port scheduling, attach the scheduler map directly to interfaces xe-0/0/31, xe-0/0/32, xe-0/0/33, and xe-0/0/34 .
ETS only: Forwarding class set (FCoE priority group)	fcoe-pg: Forwarding class fcoe Egress interfaces: xe-0/0/31, xe-0/0/32, xe-0/0/33, xe-0/0/34
ETS only: Traffic control profile	fcoe-tcp: Scheduler map fcoe-map Minimum bandwidth 3g Maximum bandwidth 100% For ETS hierarchical scheduling, attach the traffic control profile (using the output-traffic-control-profile keyword) to interfaces xe-0/0/31, xe-0/0/32, xe-0/0/33, and xe-0/0/34 .

Figure 5 on page 94 shows a block diagram of the configuration components and the configuration flow of the CLI statements used in the example.

Figure 5: PFC for FCoE Traffic Configuration Components Block Diagram



Configuration

CLI Quick Configuration

To quickly configure PFC for FCoE traffic, copy the following commands, paste them in a text file, remove line breaks, change variables and details to match your network configuration, and then copy and paste the commands into the CLI at the [edit] hierarchy level. The configuration is separated into the configuration common to ETS and direct port scheduling, and the portions of the configuration that apply only to ETS and only to port scheduling.

Common Configuration (Applies to ETS Hierarchical Scheduling and to Port Scheduling)

```
[edit class-of-service]
set classifiers ieee-802.1 fcoe-classifier forwarding-class fcoe loss-priority low code-points 011
set congestion-notification-profile fcoe-cnp input ieee-802.1 code-point 011 pfc
set interfaces xe-0/0/31 unit 0 classifiers ieee-802.1 fcoe-classifier
set interfaces xe-0/0/32 unit 0 classifiers ieee-802.1 fcoe-classifier
set interfaces xe-0/0/33 unit 0 classifiers ieee-802.1 fcoe-classifier
set interfaces xe-0/0/34 unit 0 classifiers ieee-802.1 fcoe-classifier
set interfaces xe-0/0/31 congestion-notification-profile fcoe-cnp
set interfaces xe-0/0/32 congestion-notification-profile fcoe-cnp
set interfaces xe-0/0/33 congestion-notification-profile fcoe-cnp
set interfaces xe-0/0/34 congestion-notification-profile fcoe-cnp
set schedulers fcoe-sched priority low transmit-rate 3g
set schedulers fcoe-sched shaping-rate percent 100
set scheduler-maps fcoe-map forwarding-class fcoe scheduler fcoe-sched
```

Configuration for ETS Hierarchical Scheduling

The ETS-specific portion of this example configures forwarding class set (priority group) membership, priority group CoS settings (traffic control profile), and assigns the priority group and its CoS configuration to the interfaces.

```
[edit class-of-service]
set forwarding-class-sets fcoe-pg class fcoe
set traffic-control-profiles fcoe-tcp scheduler-map fcoe-map guaranteed-rate 3g
set traffic-control-profiles fcoe-tcp shaping-rate percent 100
set interfaces xe-0/0/31 forwarding-class-set fcoe-pg output-traffic-control-profile fcoe-tcp
set interfaces xe-0/0/32 forwarding-class-set fcoe-pg output-traffic-control-profile fcoe-tcp
set interfaces xe-0/0/33 forwarding-class-set fcoe-pg output-traffic-control-profile fcoe-tcp
set interfaces xe-0/0/34 forwarding-class-set fcoe-pg output-traffic-control-profile fcoe-tcp
```

Configuration for Port Scheduling

The port-scheduling-specific portion of this example assigns the scheduler map (which sets the CoS treatment of the forwarding classes in the scheduler map) to the interfaces.

```
[edit class-of-service]
set interfaces xe-0/0/31 scheduler-map fcoe-map
set interfaces xe-0/0/32 scheduler-map fcoe-map
set interfaces xe-0/0/33 scheduler-map fcoe-map
set interfaces xe-0/0/34 scheduler-map fcoe-map
```

Common Configuration (Applies to ETS Hierarchical Scheduling and to Port Scheduling)

- Step-by-Step Procedure** To configure the ingress classifier for FCoE traffic, PFC on the FCoE traffic, apply the PFC and classifier configurations to interfaces, and configure queue scheduling, for both ETS hierarchical scheduling and port scheduling (common configuration):
1. Configure a classifier to set the loss priority and IEEE 802.1 code point assigned to the FCoE forwarding class at the ingress:


```
[edit class-of-service]
user@switch# set classifiers ieee-802.1 fcoe-classifier forwarding-class fcoe loss-priority low code-points 011
```
 2. Configure PFC on the FCoE queue by applying FCoE to the IEEE 802.1 code point 011:


```
[edit class-of-service]
user@switch# set congestion-notification-profile fcoe-cnp input ieee-802.1 code-point 011 pfc
```
 3. Apply the PFC configuration to the ingress interfaces:


```
[edit class-of-service]
user@switch# set interfaces xe-0/0/31 congestion-notification-profile fcoe-cnp
user@switch# set interfaces xe-0/0/32 congestion-notification-profile fcoe-cnp
user@switch# set interfaces xe-0/0/33 congestion-notification-profile fcoe-cnp
user@switch# set interfaces xe-0/0/34 congestion-notification-profile fcoe-cnp
```
 4. Assign the classifier to the ingress interfaces:


```
[edit class-of-service]
user@switch# set interfaces xe-0/0/31 unit 0 classifiers ieee-802.1 fcoe-classifier
user@switch# set interfaces xe-0/0/32 unit 0 classifiers ieee-802.1 fcoe-classifier
user@switch# set interfaces xe-0/0/33 unit 0 classifiers ieee-802.1 fcoe-classifier
user@switch# set interfaces xe-0/0/34 unit 0 classifiers ieee-802.1 fcoe-classifier
```
 5. Configure output scheduling for the FCoE queue:


```
[edit class-of-service]
user@switch# set schedulers fcoe-sched priority low transmit-rate 3g
user@switch# set schedulers fcoe-sched shaping-rate percent 100
```
 6. Map the FCoE forwarding class to the FCoE scheduler:


```
[edit class-of-service]
user@switch# set scheduler-maps fcoe-map forwarding-class fcoe scheduler fcoe-sched
```

ETS Hierarchical Scheduling Configuration

- Step-by-Step Procedure** To configure the forwarding class set (priority group) and priority group scheduling (in a traffic control profile), and apply the ETS hierarchical scheduling for FCoE traffic to interfaces:
1. Configure the forwarding class set for the FCoE traffic:


```
[edit class-of-service]
user@switch# set forwarding-class-sets fcoe-pg class fcoe
```
 2. Define the traffic control profile for the FCoE forwarding class set:


```
[edit class-of-service]
user@switch# set traffic-control-profiles fcoe-tcp scheduler-map fcoe-map
guaranteed-rate 3g
user@switch# set traffic-control-profiles fcoe-tcp shaping-rate percent 100
```

3. Apply the FCoE forwarding class set and traffic control profile to the egress ports:

```
[edit class-of-service]
user@switch# set interfaces xe-0/0/31 forwarding-class-set fcoe-pg
output-traffic-control-profile fcoe-tcp
user@switch# set interfaces xe-0/0/32 forwarding-class-set fcoe-pg
output-traffic-control-profile fcoe-tcp
user@switch# set interfaces xe-0/0/33 forwarding-class-set fcoe-pg
output-traffic-control-profile fcoe-tcp
user@switch# set interfaces xe-0/0/34 forwarding-class-set fcoe-pg
output-traffic-control-profile fcoe-tcp
```

Port Scheduling Configuration

Step-by-Step Procedure

To apply port scheduling for FCoE traffic to interfaces:

1. Apply the scheduler map to the egress ports:

```
[edit class-of-service]
user@switch# set interfaces xe-0/0/31 scheduler-map fcoe-map
user@switch# set interfaces xe-0/0/32 scheduler-map fcoe-map
user@switch# set interfaces xe-0/0/33 scheduler-map fcoe-map
user@switch# set interfaces xe-0/0/34 scheduler-map fcoe-map
```

Results

Display the results of the configuration (the system shows only the explicitly configured parameters; it does not show default parameters such as the **fcoe** lossless forwarding class). The results are from the ETS hierarchical scheduling configuration to show the more complex configuration. Direct port scheduling results would not show the traffic control profile or forwarding class set portions of the configuration, and would display the name of the scheduler map under each interface (instead of the names of the forwarding class set and output traffic control profile), but is otherwise the same.

```
user@switch> show configuration class-of-service
classifiers {
  ieee-802.1 fcoe-classifier {
    forwarding-class fcoe {
      loss-priority low code-points 011;
    }
  }
  traffic-control-profiles {
    fcoe-tcp {
      scheduler-map fcoe-map;
      shaping-rate percent 100;
      guaranteed-rate 3000000000;
    }
  }
  forwarding-class-sets {
    fcoe-pg {
      class fcoe;
    }
  }
}
```

```
}
congestion-notification-profile {
  fcoe-cnp {
    input {
      ieee-802.1 {
        code-point 011 {
          pfc;
        }
      }
    }
  }
}
}
}
}
interfaces {
  xe-0/0/31 {
    congestion-notification-profile fcoe-cnp;
    forwarding-class-set {
      fcoe-pg {
        output-traffic-control-profile fcoe-tcp;
      }
    }
    unit 0 {
      classifiers {
        ieee-802.1 fcoe-classifier;
      }
    }
  }
  xe-0/0/32 {
    congestion-notification-profile fcoe-cnp;
    forwarding-class-set {
      fcoe-pg {
        output-traffic-control-profile fcoe-tcp;
      }
    }
    unit 0 {
      classifiers {
        ieee-802.1 fcoe-classifier;
      }
    }
  }
  xe-0/0/33 {
    congestion-notification-profile fcoe-cnp;
    forwarding-class-set {
      fcoe-pg {
        output-traffic-control-profile fcoe-tcp;
      }
    }
    unit 0 {
      classifiers {
        ieee-802.1 fcoe-classifier;
      }
    }
  }
  xe-0/0/34 {
    congestion-notification-profile fcoe-cnp;
    forwarding-class-set {
      fcoe-pg {
```

```

        output-traffic-control-profile fcoe-tcp;
    }
}
unit 0 {
    classifiers {
        ieee-802.1 fcoe-classifier;
    }
}
}
}
scheduler-maps {
    fcoe-map {
        forwarding-class fcoe scheduler fcoe-sched;
    }
}
schedulers {
    fcoe-sched {
        transmit-rate 3000000000;
        shaping-rate percent 100;
        priority low;
    }
}
}

```



TIP: To quickly configure the interfaces, issue the **load merge terminal** command and then copy the hierarchy and paste it into the switch terminal window.

Verification

To verify that the PFC configuration for FCoE traffic components has been created and is operating properly, perform these tasks:

- [Verifying That Priority-Based Flow Control Has Been Enabled on page 99](#)
- [Verifying the Ingress Interface PFC Configuration on page 100](#)

Verifying That Priority-Based Flow Control Has Been Enabled

Purpose Verify that PFC is enabled on the FCoE queue to enable lossless transport.

Action List the congestion notification profiles using the operational mode command **show class-of-service congestion-notification**:

```

user@switch> show class-of-service congestion-notification
Type: Input, Name: fcoe-cnp, Index: 51697
Cable Length: 100 m

```

Priority	PFC	MRU
000	Disabled	
001	Disabled	
010	Disabled	
011	Enabled	2500
100	Disabled	
101	Disabled	

110	Disabled
111	Disabled
Type: Output	
Priority	Flow-Control-Queues
000	0
001	1
010	2
011	3
100	4
101	5
110	6
111	7

Meaning The **show class-of-service congestion-notification** operational command lists all of the congestion notification profiles and which IEEE 802.1p code points have PFC enabled. The command output shows that PFC is enabled on code point **011** for the **fcoe-cnp** congestion notification profile.

The command also shows the default cable length (100 meters), the default maximum receive unit (2500 bytes), and the default mapping of priorities to output queues because this example does not include configuring these options.

Verifying the Ingress Interface PFC Configuration

Purpose Verify that the classifier **fcoe-classifier** and the congestion notification profile **fcoe-cnp** are configured on ingress interfaces **xe-0/0/31**, **xe-0/0/32**, **xe-0/0/33**, and **xe-0/0/34**.

Action List the ingress interfaces using the operational mode command **show configuration class-of-service interfaces**:

```
user@switch> show configuration class-of-service interfaces xe-0/0/31
congestion-notification-profile fcoe-cnp;
unit 0 {
    classifiers {
        ieee-802.1 fcoe-classifier;
    }
}

user@switch> show configuration class-of-service interfaces xe-0/0/32
congestion-notification-profile fcoe-cnp;
unit 0 {
    classifiers {
        ieee-802.1 fcoe-classifier;
    }
}

user@switch> show configuration class-of-service interfaces xe-0/0/33
congestion-notification-profile fcoe-cnp;
unit 0 {
```

```

        classifiers {
            ieee-802.1 fcoe-classifier;
        }
    }

user@switch> show configuration class-of-service interfaces xe-0/0/34
congestion-notification-profile fcoe-cnp;
unit 0 {
    classifiers {
        ieee-802.1 fcoe-classifier;
    }
}

```

Meaning The **show configuration class-of-service interfaces** commands list the congestion notification profile that is mapped to the interface (**fcoe-cnp**) and the IEEE 802.1p classifier associated with the interface (**fcoe-classifier**).

Related Documentation

- [Example: Configuring CoS Hierarchical Port Scheduling \(ETS\)](#)
- [Configuring CoS PFC \(Congestion Notification Profiles\)](#)
- [Understanding CoS Flow Control \(Ethernet PAUSE and PFC\) on page 81](#)

Understanding DCBX

Data Center Bridging Capability Exchange protocol (DCBX) is an extension of Link Layer Data Protocol (LLDP). If you disable LLDP on an interface, that interface cannot run DCBX. If you attempt to enable DCBX on an interface on which LLDP is disabled, the configuration commit operation fails. Data center bridging (DCB) devices use DCBX to exchange configuration information with directly connected peers.



Video: [What is DCBX Protocol?](#)

This topic describes:

- [DCBX Basics on page 101](#)
- [DCBX Modes and Support on page 103](#)
- [DCBX Attribute Types on page 105](#)
- [DCBX Application Protocol TLV Exchange on page 107](#)
- [DCBX and PFC on page 108](#)
- [DCBX and ETS on page 108](#)

DCBX Basics

DCBX can:

- Discover the DCB capabilities of peers.
- Detect DCB feature misconfiguration or mismatches between peers.

- Configure DCBX features on peers.

You can configure DCBX operation for priority-based flow control (PFC), Layer 2 and Layer 4 applications such as FCoE and iSCSI, and ETS. DCBX is enabled or disabled on a per-interface basis.



NOTE: The Juniper Networks QFX10000 does not support enhanced transmission selection (ETS) hierarchical scheduling. Use port scheduling to manage bandwidth on QFX10000 switches.

By default, for PFC and ETS, DCBX automatically negotiates administrative state and configuration with each interface's connected peer. To enable DCBX negotiation for applications, you must configure the applications, map them to IEEE 802.1p code points in an application map, and apply the application map to interfaces.

The FCoE application only needs to be included in an application map when you want an interface to exchange type, length, and values (TLVs) for other applications in addition to FCoE. If FCoE is the only application you want an interface to advertise, then you do not need to use an application map. For ETS, DCBX pushes the switch configuration to peers if they are set to learn the configuration from the switch (unless you disable sending the ETS recommendation TLV on interfaces in IEEE DCBX mode).

You can override the default behavior for PFC, for ETS, or for all applications mapped to an interface by turning off autonegotiation to force an interface to enable or disable that feature. You can also disable DCBX autonegotiation for applications on an interface by excluding those applications from the application map you apply to that interface or by deleting the application map from the interface.

The default autonegotiation behavior for applications that are mapped to an interface is:

- DCBX is enabled on the interface if the connected peer device also supports DCBX.
- DCBX is disabled on the interface if the connected peer device does not support DCBX.

During negotiation of capabilities, the switch can push the PFC configuration to an attached peer if the peer is configured as “willing” to learn the PFC configuration from other peers. The Juniper Networks switch does not support self autoprovisioning and does not change its configuration during autonegotiation to match the peer configuration. (The Juniper switch is not “willing” to learn the PFC configuration from peers.)



NOTE: When a port with DCBX enabled begins to exchange type, length, and value (TLV) entries, optional LLDP TLVs on that port are not advertised to neighbors, so that the switch can interoperate with a wider variety of converged network adapters (CNAs) and Layer 2 switches that support DCBX.

DCBX Modes and Support

This section describes DCBX support:

- [DCBX Modes \(Versions\) on page 103](#)
- [Autonegotiation on page 105](#)
- [CNA Support for DCBX Modes on page 105](#)
- [Interface Support for DCBX on page 105](#)

DCBX Modes (Versions)

The two most common DCBX modes are supported:

- IEEE DCBX—The newest DCBX version. Different TLVs have different subtypes (for example, the subtype for the ETS configuration TLV is 9); the IEEE DCBX Organizationally Unique Identifier (OUI) is 0x0080c2.
- DCBX version 1.01—The Converged Enhanced Ethernet (CEE) version of DCBX. It has a subtype of 2 and an OUI of 0x001b21.

IEEE DCBX and DCBX version 1.01 differ mainly in frame format. DCBX version 1.01 uses one TLV that includes all DCBX attribute information, which is sent as sub-TLVs. IEEE DCBX uses a unique TLV for each DCB attribute.



NOTE: The switch does not support pre-CEE (pre-DCB) DCBX versions. Unsupported older versions of DCBX have a subtype of 1 and an OUI of 0x001b21. The switch drops LLDP frames that contain pre-CEE DCBX TLVs.

[Table 9 on page 103](#) summarizes the differences between IEEE DCBX and DCBX version 1.01, including show command output:

Table 9: Summary of Differences Between IEEE DCBX and DCBX Version 1.01

Characteristic	IEEE DCBX	DCBX Version 1.01
OUI	0x0080c2	0x001b21
Frame Format	Sends a separate, unique TLV for each DCBX attribute. For example, IEEE DCBX uses separate TLVs for ETS, PFC, and each application. Configuration and Recommendation information is sent in different TLVs	Sends one TLV that includes all DCBX attribute information organized in sub-TLVs. The “willing” bit determines whether or not an interface can change its configuration to match the connected peer.
Symmetric/asymmetric configuration with peer	Asymmetric or symmetric	Symmetric only

Table 9: Summary of Differences Between IEEE DCBX and DCBX Version 1.01 (continued)

Characteristic	IEEE DCBX	DCBX Version 1.01
Differences in the show dcbx interface interface-name operational command	<ul style="list-style-type: none"> Synchronization information is not shown because symmetric configuration is not required. Operational state information is not shown because the operational states do not have to be symmetric. TLV type is shown because unique TLVs are sent for each DCBX attribute. ETS peer Configuration TLV and Recommendation TLV information is shown separately because they are different TLVs. 	<ul style="list-style-type: none"> Synchronization information is shown because symmetric configuration is required. Operational state information is shown because the operational states do have to be symmetric. TLV type is not shown because one TLV is used for all attribute information. Recommendation TLV is not sent (DCBX Version 1.01 uses the "willing" bit to determine whether or not an interface uses the peer interface configuration).

For more information about how each DCBX mode exchanges TLVs, see the following specifications:

- For DCBX version 1.01—<http://www.ieee802.org/1/files/public/docs2008/az-wedlar-dcb-capability-exchange-discovery-protocol-108-v1.01.pdf>
- For IEEE DCBX—<http://www.ieee802.org/1/files/private/az-drafts/d2/802-1az-d2-4.pdf>



NOTE: As of Junos OS Release 12.2, this document is located in a private area of the IEEE website, and access requires a password from the IEEE organization. If you are not an IEEE member, you might not be able to access this document until it moves to the public area of the IEEE website.

You can configure interfaces to use the following DCBX modes:

- IEEE DCBX—The interface uses IEEE DCBX regardless of the configuration on the connected peer.
- DCBX version 1.01—The interface uses DCBX version 1.01 regardless of the configuration on the connected peer.
- Autonegotiation—The interface automatically negotiates with the connected peer to determine the DCBX version the peers use. Autonegotiation is the default DCBX mode.

If you configure a DCBX mode on an interface, the interface ignores DCBX protocol data units (PDUs) it receives from the connected peer if the PDUs do not match the DCBX version configured on the interface. For example, if you configure an interface to use IEEE DCBX and the connected peer sends DCBX version 1.01 LLDP PDUs, the interface ignores the version 1.01 PDUs. If you configure an interface to use DCBX version 1.01 and the peer sends IEEE DCBX LLDP PDUs, the interface ignores the IEEE DCBX PDUs.



NOTE: On interfaces that use the IEEE DCBX mode, the `show dcbx neighbors interface interface-name` operational command does not include application, PFC, or ETS operational state in the output.

Autonegotiation

Autonegotiation is the default DCBX mode. Each interface automatically negotiates with its connected peer to determine the DCBX version that both interfaces use to exchange DCBX information.

When an interface connects to its peer interface, the interface advertises IEEE DCBX TLVs to the peer. If the interface receives one IEEE DCBX PDU from the peer, the interface sets the DCBX mode as IEEE DCBX. If the interface receives three DCBX version 1.01 TLVs from the peer, the interface sets DCBX version 1.01 as the DCBX mode.

Autonegotiation works slightly differently on standalone switches compared to QFabric systems:

- Standalone switches—When an interface connects to its peer interface, the interface advertises IEEE DCBX TLVs to the peer. If the interface receives an IEEE DCBX TLV from the peer, the interface sets IEEE DCBX as the DCBX mode. If the interface receives three consecutive DCBX version 1.01 TLVs from the peer, the interface sets DCBX version 1.01 as the DCBX mode.
- QFabric system—When an interface connects to its peer interface, the interface advertises DCBX version 1.01 TLVs to the peer. If the interface receives an IEEE DCBX TLVs from the peer, the interface sets IEEE DCBX as the DCBX mode. If the interface receives three consecutive DCBX version 1.01 TLVs from the peer, the interface retains DCBX version 1.01 as the DCBX mode.



NOTE: If the link flaps or the LLDP process restarts, the interface starts the autonegotiation process again. The interface does not use the last received DCBX communication mode.

CNA Support for DCBX Modes

Different CNA vendors support different versions and capabilities of DCBX. The DCBX configuration you use on switch interfaces depends on the DCBX features that the CNAs in your network support.

Interface Support for DCBX

You can configure DCBX on 10-Gigabit Ethernet interfaces and on link aggregation group (LAG) interfaces whose member interfaces are all 10-Gigabit Ethernet interfaces.

DCBX Attribute Types

DCBX has three attribute types:

- **Informational**—These attributes are exchanged using LLDP, but do not affect DCBX state or operation; they only communicate information to the peer. For example, application priority TLVs are informational TLVs.
- **Asymmetric**—The values for these types of attributes do not have to be the same on the connected peer interfaces. Peers exchange asymmetric attributes when the attribute values can differ on each peer interface. The peer interface configurations might match or they might differ. For example, ETS Configuration and Recommendation TLVs are asymmetric TLVs.
- **Symmetric**—The intention is that the values for these types of attributes should be the same on both of the connected peer interfaces. Peer interfaces exchange symmetric attributes to ensure symmetric DCBX configuration for those attributes. For example, PFC Configuration TLVs are symmetric TLVs.

The following sections describe asymmetric and symmetric DCBX attributes:

- [Asymmetric Attributes on page 106](#)
- [Symmetric Attributes on page 106](#)

Asymmetric Attributes

DCBX passes asymmetric attributes between connected peer interfaces to communicate parameter information about those attributes (features). The resulting configuration for an attribute might be different on each peer, so the parameters configured on one interface might not match the parameters on the connected peer interface.

There are two types of asymmetric attribute TLVs:

- **Configuration TLV**—Configuration TLVs communicate the current operational state and the state of the “willing” bit. The “willing” bit communicates whether or not the interface is willing to accept and use the configuration from the peer interface. If an interface is “willing,” the interface uses the configuration it receives from the peer interface. (The peer interface configuration can override the configuration on the “willing” interface.) If an interface is “not willing”, the configuration on the interface cannot be overridden by the peer interface configuration.
- **Recommendation TLV**—Recommendation TLVs communicate the parameters the interface recommends that the connected peer interface should use. When an interface sends a Recommendation TLV, if the connected peer is “willing,” the connected peer changes its configuration to match the parameters in the Recommendation TLV.

Symmetric Attributes

DCBX passes symmetric attributes between connected peer interfaces to communicate parameter information about those attributes (features), with the objective that both interfaces should use the same configuration. The intent is that the parameters configured on one interface should match the parameters on the connected peer interface.

There is one type of symmetric attribute TLV, the Configuration TLV. As with asymmetric attributes, symmetric attribute Configuration TLVs communicate the current operational state and the state of the “willing” bit. “Willing” interfaces use the peer interface parameter

values for the attribute. (The attribute configuration of the peer overrides the configuration on the “willing” interface.)

DCBX Application Protocol TLV Exchange

DCBX advertises the switch's capabilities for Layer 2 applications such as FCoE and Layer 4 applications such as iSCSI:

- [Application Protocol TLV Exchange on page 107](#)
- [FCoE Application Protocol TLV Exchange on page 107](#)
- [Disabling Application Protocol TLV Exchange on page 108](#)

Application Protocol TLV Exchange

For all applications, DCBX advertises the application's state and IEEE 802.1p code points on the interfaces to which the application is mapped. If an application is not mapped to an interface, that interface does not advertise the application's TLVs. There is an exception for FCoE application protocol TLV exchange when FCoE is the only application you want DCBX to advertise on an interface.

FCoE Application Protocol TLV Exchange

Protocol TLV exchange for the FCoE application depends on whether FCoE is the only application you want the interface to advertise or whether you want the interface to exchange other application TLVs in addition to FCoE TLVs.

If FCoE is the only application you want DCBX to advertise on an interface, DCBX exchanges FCoE application protocol TLVs by default if the interface:

- Carries FCoE traffic (traffic mapped by CoS configuration to the FCoE forwarding class)
- Has a congestion notification profile with PFC enabled on the FCoE priority (IEEE 802.1p code point)
- Does *not* have an application map



NOTE: If no CoS configuration for FCoE is mapped to an interface, that interface does not exchange FCoE application protocol TLVs.

If you want DCBX to advertise FCoE and other applications on an interface, you must specify all of the applications, including FCoE, in an application map, and apply the application map to the desired interfaces.



NOTE: If an application map is applied to an interface, the FCoE application must be explicitly configured in the application map, or the interface does not exchange FCoE TLVs.

When DCBX advertises the FCoE application, it advertises the FCoE state and IEEE 802.1p code points. If a peer device connected to a switch interface does not support FCoE,

DCBX uses autonegotiation to mark the interface as “FCoE down,” and FCoE is disabled on that interface.

Disabling Application Protocol TLV Exchange

To disable DCBX application protocol exchange for all applications on an interface, issue the **set protocols dcbx interface *interface-name* applications no-auto-negotiation** command.

You can also disable DCBX application protocol exchange for applications on an interface by deleting the application map from the interface, or by deleting a particular application from the application map. However, when you delete an application from an application map, the application protocol is no longer exchanged on any interface which uses that application map.

DCBX and PFC

After you enable PFC on a switch interface, DCBX uses autonegotiation to control the operational state of the PFC functionality.

If the peer device connected to the interface supports PFC and is provisioned compatibly with the switch, DCBX sets the PFC operational state to enabled. If the peer device connected to the interface does not support PFC or is not provisioned compatibly with the switch, DCBX sets the operational state to disabled. (PFC must be symmetrical.)

If the peer advertises that it is “willing” to learn its PFC configuration from the switch, DCBX pushes the switch’s PFC configuration to the peer and does not check the peer’s administrative state.

You can manually override DCBX control of the PFC operational state on a per-interface basis by disabling autonegotiation. If you disable autonegotiation on an interface on which you have configured PFC, then PFC is enabled on that interface regardless of the peer configuration. To disable PFC on an interface, do not configure PFC on that interface.

DCBX and ETS

This section describes:

- [Default DCBX ETS Advertisement on page 108](#)
- [ETS Advertisement and Peer Configuration on page 109](#)
- [ETS Recommendation TLV on page 109](#)

Default DCBX ETS Advertisement

If you do not configure ETS on an interface, the switch automatically creates a default priority group that contains all of the priorities (forwarding classes, which represent output queues) and assigns 100 percent of the port output bandwidth to that priority group. The default priority group is transparent. It does not appear in the configuration and is used for DCBX advertisement. DCBX advertises the default priority group, its priorities, and the assigned bandwidth.

If you configure ETS on an interface, DCBX advertises:

- Each priority group on the interface

- The priorities in each priority group
- The bandwidth properties of each priority group and priority

Any priority on that interface that is not part of an explicitly configured priority group (forwarding class set) is assigned to the automatically generated default priority group and receives no bandwidth. If you configure ETS on an interface, every forwarding class (priority) on that interface for which you want to forward traffic must belong to a forwarding class set (priority group).

ETS Advertisement and Peer Configuration

DCBX does not control the switch's ETS (hierarchical scheduling) operational state. If the connected peer is configured as "willing," DCBX pushes the switch's ETS configuration to the switch's peers if the ETS Recommendation TLV is enabled (it is enabled by default). If the peer does not support ETS or is not consistently provisioned with the switch, DCBX does not change the ETS operational state on the switch. The ETS operational state remains enabled or disabled based only on the switch hierarchical scheduling configuration and is enabled by default.

When ETS is configured, DCBX advertises the priority groups, the priorities in the priority groups, and the bandwidth configuration for the priority groups and priorities. Any priority (essentially a forwarding class or queue) that is not part of a priority group has no scheduling properties and receives no bandwidth.

You can manually override whether DCBX advertises the ETS state to the peer on a per-interface basis by disabling autonegotiation. This does not affect the ETS state on the switch or on the peer, but it does prevent the switch from sending the Recommendation TLV or the Configuration TLV to the connected peer. To disable ETS on an interface, do not configure priority groups (forwarding class sets) on the interface.

ETS Recommendation TLV

The ETS Recommendation TLV communicates the ETS settings that the switch wants the connected peer interface to use. If the peer interface is "willing," it changes its configuration to match the configuration in the ETS Recommendation TLV. By default, the switch interfaces send the ETS Recommendation TLV to the peer. The settings communicated are the egress ETS settings defined by configuring hierarchical scheduling on the interface.

We recommend that you use the same ETS settings on the connected peer that you use on the switch interface and that you leave the ETS Recommendation TLV enabled. However, on interfaces that use IEEE DCBX as the DCBX mode, if you want an asymmetric configuration between the switch interface and the connected peer, you can disable the ETS Recommendation TLV by including the **no-recommendation-tlv** statement at the **[edit protocols dcbx interface *interface-name* enhanced-transmission-selection]** hierarchy level.



NOTE: You can disable the ETS Recommendation TLV only when the DCBX mode on the interface is IEEE DCBX. Disabling the ETS Recommendation TLV has no effect if the DCBX mode on the interface is DCBX version 1.01. (IEEE DCBX uses separate application attribute TLVs, but DCBX version 1.01 sends all application attributes in the same TLV and uses sub-TLVs to separate the information.)

If you disable the ETS Recommendation TLV, the switch still sends the ETS Configuration TLV to the connected peer. The result is that the connected peer is informed about the switch DCBX ETS configuration, but even if the peer is “willing,” the peer does not change its configuration to match the switch configuration. This is asymmetric configuration—the two interfaces can have different parameter values for the ETS attribute.

For example, if you want a CNA connected to a switch interface to have different bandwidth allocations than the switch ETS configuration, you can disable the ETS Recommendation TLV and configure the CNA for the desired bandwidth. The switch interface and the CNA exchange configuration parameters, but the CNA does not change its configuration to match the switch interface configuration.

**Related
Documentation**

- [Understanding DCBX Application Protocol TLV Exchange on page 116](#)
- [Understanding DCB Features and Requirements on page 78](#)
- [Understanding CoS Flow Control \(Ethernet PAUSE and PFC\) on page 81](#)
- [Understanding CoS Hierarchical Port Scheduling \(ETS\)](#)
- [Understanding CoS Port Schedulers on QFX Switches](#)
- [Understanding FCoE on page 26](#)
- [Configuring the DCBX Mode on page 111](#)
- [Configuring DCBX Autonegotiation on page 112](#)
- [Disabling the ETS Recommendation TLV on page 115](#)
- [Example: Configuring DCBX Application Protocol TLV Exchange on page 123](#)

Configuring the DCBX Mode

You can configure the DCBX mode that an interface uses to communicate with the connected peer. Three DCBX modes are supported:

- **Autonegotiation**—The interface negotiates with the connected peer to determine the DCBX mode. This is the default DCBX mode.
- **IEEE DCBX**—The interface uses IEEE DCBX type, length, and value (TLV) to exchange DCBX information with the connected peer. QFX3500 Node devices come up with IEEE DCBX enabled by default and then autonegotiate with the connected peer to determine the final DCBX mode.
- **DCBX Version 1.01**—The interface uses Converged Enhanced Ethernet (CEE) DCBX version 1.01 TLVs to exchange DCBX information with the connected peer. QFabric system Node devices other than QFX3500 switches come up with DCBX version 1.01 enabled by default and then autonegotiate with the connected peer to determine the final DCBX mode.



NOTE: Pre-CEE (pre-DCB) versions of DCBX such as DCBX version 1.00 are not supported. If an interface receives an LLDP frame with pre-CEE DCBX TLVs, the system drops the frame.

Configure the DCBX mode by specifying the mode for one interface or for all interfaces.

- To configure the DCBX mode, specify the interface and the mode:

```
[edit protocols dcbx]
user@switch# set interface interface-name mode (auto-negotiate | ieee-dcbx |
dcbx-version-1.01)
```

For example, to configure DCBX version 1.01 on interface **xe-0/0/21**:

```
user@switch# set protocols dcbx interface xe-0/0/21 mode dcbx-version-1.01
```

To configure IEEE DCBX on all interfaces:

```
user@switch# set protocols dcbx interface all mode ieee-dcbx
```

Related Documentation

- [Configuring DCBX Autonegotiation on page 112](#)
- [Disabling the ETS Recommendation TLV on page 115](#)
- [Understanding DCBX on page 101](#)
- [Understanding DCBX Application Protocol TLV Exchange on page 116](#)
- [show dcbx neighbors on page 169](#)

Configuring DCBX Autonegotiation

Data Center Bridging Capability Exchange protocol (DCBX) discovers the data center bridging (DCB) capabilities of peers by exchanging feature configuration information. DCBX also detects feature misconfiguration and mismatches, and can configure DCB on peers. DCBX is an extension of the Link Layer Discovery Protocol (LLDP), and LLDP must remain enabled on every interface for which you want to use DCBX. If you attempt to enable DCBX on an interface on which LLDP is disabled, the configuration commit operation fails.



NOTE: LLDP and DCBX are enabled by default on all interfaces.

The switch supports DCBX autonegotiation for:

- Priority-based flow control (PFC) configuration
- Layer 2 and Layer 4 applications such as Fibre Channel over Ethernet (FCoE) and Internet Small Computer System Interface (iSCSI)
- Enhanced transmission selection (ETS) advertisement

DCBX autonegotiation is configured on a per-interface basis for each supported feature or application. The PFC and application DCBX exchanges use autonegotiation by default. The default autonegotiation behavior is:

- DCBX is enabled on the interface if the connected peer device also supports DCBX.
- DCBX is disabled on the interface if the connected peer device does not support DCBX.

You can override the default behavior for each feature by turning off autonegotiation to force an interface to enable or disable the feature.

Autonegotiation of ETS means that when ETS is enabled on an interface (priority groups are configured), the interface advertises its ETS configuration to the peer device. In this case, priorities (forwarding classes) that are not part of a priority group (forwarding class set) receive no bandwidth and are advertised in an automatically generated default forwarding class. If ETS is not enabled on an interface (no priority groups are configured), all of the priorities are advertised in one automatically generated default priority group that receives 100 percent of the port bandwidth.

Disabling ETS autonegotiation prevents the interface from sending the Recommendation TLV or the Configuration TLV to the connected peer.

On interfaces that use IEEE DCBX mode to exchange DCBX parameters, you can disable autonegotiation of the ETS Recommendation TLV to the peer if you want an asymmetric ETS configuration between the peers. DCBX still exchanges the ETS Configuration TLV if you disable the ETS Recommendation TLV.

Autonegotiation of PFC means that when PFC is enabled on an interface, if the peer device connected to the interface supports PFC and is provisioned compatibly with the

switch, DCBX sets the PFC operational state to enabled. If the peer device connected to the interface does not support PFC or is not provisioned compatibly with the switch, DCBX sets the operational state to disabled.

In addition, if the peer advertises that it is “willing” to learn its PFC configuration from the switch, DCBX pushes the switch’s PFC configuration to the peer and does not check the peer’s administrative state. The switch does not learn PFC configuration from peers (the switch does not advertise its state as “willing”).

Disabling PFC autonegotiation prevents the interface from exchanging PFC configuration information with the peer. It forces the interface to enable PFC if PFC is configured on the interface or to disable PFC if PFC is not configured on the interface. If you disable PFC autonegotiation, the assumption is that the peer is also configured manually.

Autonegotiation of applications depends on whether or not you apply an application map to an interface. If you apply an application map to an interface, the interface autonegotiates DCBX for each application in the application map. PFC must be enabled on the FCoE priority (the FCoE IEEE 802.1p code point) for the interface to advertise the FCoE application. The interface only advertises applications that are included in the application map.

For example, if you apply an application map to an interface and the application map does not include the FCoE application, then that interface does not perform DCBX advertisement of FCoE.

If you do not apply an application map to an interface, DCBX does not advertise applications on that interface, with the exception of FCoE, which is handled differently than other applications.



NOTE: If you do not apply an application map to an interface, the interface performs autonegotiation of FCoE if the interface carries traffic in the FCoE forwarding class and also has PFC enabled on the FCoE priority. On such interfaces, if DCBX detects that the peer device connected to the interface supports FCoE, the switch advertises its FCoE capability and IEEE 802.1p code point on that interface. If DCBX detects that the peer device connected to the interface does not support FCoE, DCBX marks that interface as “FCoE down” and disables FCoE on the interface.

When DCBX marks an interface as “FCoE down,” the behavior of the switch depends on how you use it in the network:

- When the switch acts as an FCoE transit switch, the interface drops all of the FIP packets it receives. In addition, FIP packets received from an FCoE forwarder (FCF) are not forwarded to interfaces marked as “FCoE down.”
- When the switch acts as an FCoE-FC gateway (only switches that support native Fibre Channel interfaces), it does not send or receive FCoE Initialization Protocol (FIP) packets.

Disabling autonegotiation prevents the interface from exchanging application information with the peer. In this case, the assumption is that the peer is also configured manually.

To disable DCBX autonegotiation of PFC, applications (including FCoE), and ETS using the CLI:

1. Turn off autonegotiation for PFC.

```
[edit]
user@switch# set protocols dcbx interface interface-name priority-flow-control
no-auto-negotiation
```

2. Turn off autonegotiation for applications.

```
[edit]
user@switch# set protocols dcbx interface interface-name applications no-auto-negotiation
```

3. Turn off autonegotiation for ETS.

```
[edit]
user@switch# set protocols dcbx interface interface-name enhanced-transmission-selection
no-auto-negotiation
```

To disable autonegotiation of the ETS Recommendation TLV so that DCBX exchanges only the ETS Configuration TLV:

- [edit protocols dcbx interface *interface-name*]
user@switch# set enhanced-transmission-selection no-recommendation-tlv

**Related
Documentation**

- [Example: Configuring DCBX Application Protocol TLV Exchange on page 123](#)
- [Example: Configuring CoS PFC for FCoE Traffic on page 92](#)
- [Disabling the ETS Recommendation TLV on page 115](#)
- [Understanding DCBX Application Protocol TLV Exchange on page 116](#)

Disabling the ETS Recommendation TLV

The enhanced transmission selection (ETS) Recommendation TLV communicates the ETS settings that the switch wants the connected peer interface to use. If the peer interface is “willing,” the peer interface changes its configuration to match the configuration in the ETS Recommendation TLV. By default, the switch interfaces send the ETS Recommendation TLV to the peer. The settings communicated are the egress ETS settings defined by configuring hierarchical scheduling on the interface.

We recommend that you use the same ETS settings on the connected peer that you use on the switch interface and that you leave the ETS Recommendation TLV enabled. However, on interfaces that use IEEE DCBX as the DCBX mode, if you want an asymmetric configuration between the switch interface and the connected peer, you can disable the ETS Recommendation TLV.



NOTE: Disabling the ETS Recommendation TLV on interfaces that use DCBX version 1.01 as the DCBX mode has no effect and does not change DCBX behavior.

If you disable the ETS Recommendation TLV, the switch still sends the ETS Configuration TLV to the connected peer. The result is that the connected peer is informed about the switch DCBX ETS configuration, but even if the peer is “willing,” the peer does not change its configuration to match the switch configuration. This is asymmetric configuration—the two interfaces can have different parameter values for the ETS attribute.

To disable the ETS Recommendation TLV:

- [edit protocols dcbx interface *interface-name*]
user@switch# **set enhanced-transmission-selection no-recommendation-tlv**

Related Documentation

- [Configuring the DCBX Mode on page 111](#)
- [Configuring DCBX Autonegotiation on page 112](#)
- [Understanding DCBX on page 101](#)
- [Understanding Data Center Bridging Capability Exchange Protocol for EX Series Switches](#)

Understanding DCBX Application Protocol TLV Exchange

Data Center Bridging Capability Exchange protocol (DCBX) discovers the data center bridging (DCB) capabilities of connected peers. DCBX also advertises the capabilities of applications on interfaces by exchanging application protocol information through application type, length, and value (TLV) elements. DCBX is an extension of Link Layer Discovery Protocol (LLDP). LLDP must remain enabled on every interface on which you want to use DCBX.



NOTE: LLDP and DCBX are enabled by default on all interfaces.

Setting up application protocol exchange consists of:

- Defining applications
- Mapping the applications to IEEE 802.1p code points in an *application map*
- Configuring classifiers to prioritize incoming traffic and map the incoming traffic to the application by the traffic code points
- Applying the application maps and classifiers to interfaces

You need to explicitly define the applications that you want an interface to advertise. The FCoE application is a special case (see [“Applications” on page 116](#)) and only needs to be defined on an interface if you want DCBX to exchange application protocol TLVs for other applications in addition to FCoE on that interface.

You also need to explicitly map all of the defined applications that you want an interface to advertise to IEEE 802.1p code points in an application map. The FCoE application is a special case that only requires inclusion in an application map when you want an interface to use DCBX for other applications in addition to FCoE, as described later in this topic (see [“Application Maps” on page 117](#)).

This topic describes:

- [Applications on page 116](#)
- [Application Maps on page 117](#)
- [Classifying and Prioritizing Application Traffic on page 118](#)
- [Enabling Interfaces to Exchange Application Protocol Information on page 119](#)
- [Disabling DCBX Application Protocol Exchange on page 119](#)

Applications

Before an interface can exchange application protocol information, you need to define the applications that you want to advertise. The exception is the FCoE application. If FCoE is the only application that you want the interface to advertise, then you do not need to define the FCoE application. You need to define the FCoE application only if you want interfaces to advertise other applications in addition to FCoE.



NOTE: If FCoE is the only application that you want DCBX to advertise on an interface, DCBX exchanges FCoE application protocol TLVs by default if the interface:

- Carries FCoE traffic (traffic mapped by CoS configuration to the FCoE forwarding class and applied to the interface)
- Has a congestion notification profile with PFC enabled on the FCoE priority (IEEE 802.1p code point)
- Does *not* have an application map

If you apply an application map to an interface, then all applications that you want DCBX to advertise must be defined and configured in the application map, including the FCoE application.

If no CoS configuration for FCoE is mapped to an interface, that interface does not exchange FCoE application protocol TLVs.

You can define:

- Layer 2 applications by EtherType
- Layer 4 applications by a combination of protocol (TCP or UDP) and destination port number

The EtherType is a two-octet field in the Ethernet frame that denotes the protocol encapsulated in the frame. For a list of common EtherTypes, see <http://standards.ieee.org/develop/regauth/ethertype/eth.txt> on the IEEE standards organization website. For a list of port numbers and protocols, see the *Service Name and Transport Protocol Port Number Registry* at <http://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xml> on the Internet Assigned Numbers Authority (IANA) website.

You must explicitly define each application that you want to advertise, except FCoE. The FCoE application is defined by default (EtherType 0x8906).

Application Maps

An application map maps defined applications to one or more IEEE 802.1p code points. Each application map contains one or more applications. DCBX includes the configured application code points in the protocol TLVs exchanged with the connected peer.

To exchange protocol TLVs for an application, you must include the application in an application map. The FCoE application is a special case:

- If you want DCBX to exchange application protocol TLVs for more than one application on a particular interface, you must configure the applications, define an application map to map the applications to code points, and apply the application map to the interface. In this case, you must also define the FCoE application and add it to the application map.

This is the same process and treatment required for all other applications. In addition, for DCBX to exchange FCoE application TLVs, you must enable priority-based flow control (PFC) on the FCoE priority (the FCoE IEEE 802.1p code point) on the interface.

- If FCoE is the only application that you want DCBX to advertise on an interface, then you do not need to configure an application map and apply it to the interface. By default, when an interface has no application map, and the interface carries traffic mapped to the FCoE forwarding class, and PFC is enabled on the FCoE priority, the interface advertises FCoE TLVs (autonegotiation mode). DCBX exchanges FCoE application protocol TLVs by default until you apply an application map to the interface, remove the FCoE traffic from the interface (you can do this by removing the or editing the classifier for FCoE traffic), or disable PFC on the FCoE priority.

If you apply an application map to an interface that did not have an application map and was exchanging FCoE application TLVs, and you do not include the FCoE application in the application map, the interface stops exchanging FCoE TLVs. Every interface that has an application map must have FCoE included in the application map (and PFC enabled on the FCoE priority) in order for DCBX to exchange FCoE TLVs.

Mapping an application to code points does two things:

- Maps incoming traffic with the same code points to that application
- Allows you to configure classifiers that map incoming application traffic, by code point, to a forwarding class and a loss priority, in order to apply class of service (CoS) to application traffic and prioritize application traffic

You apply an application map to an interface to enable DCBX application protocol exchange on that interface for each application specified in the application map. All of the applications that you want an interface to advertise must be configured in the application map that you apply to the interface, with the previously noted exception for the FCoE application when FCoE is the only application for which you want DCBX to exchange protocol TLVs on an interface.

Classifying and Prioritizing Application Traffic

When traffic arrives at an interface, the interface classifies the incoming traffic based on its code points. Classifiers map code points to loss priorities and forwarding classes. The loss priority prioritizes the traffic. The forwarding class determines the traffic output queue and CoS service level.

When you map an application to an IEEE 802.1p code point in an application map and apply the application map to an interface, incoming traffic on the interface that matches the application code points is mapped to the appropriate application. The application receives the loss priority and the CoS associated with the forwarding class for those code points, and is placed in the output queue associated with the forwarding class.

You can use the default classifier or you can configure a classifier to map the application code points defined in the application map to forwarding classes and loss priorities.

Enabling Interfaces to Exchange Application Protocol Information

Each interface with the **fcoe** forwarding class and PFC enabled on the FCoE code point is enabled for FCoE application protocol exchange by default until you apply an application map to the interface. If you apply an application map to an interface and you want that interface to exchange FCoE application protocol TLVs, you must include the FCoE application in the application map. (In all cases, to achieve lossless transport, you must also enable PFC on the FCoE code point or code points.)

Except when FCoE is the only protocol you want DCBX to advertise on an interface, interfaces on which you want to exchange application protocol TLVs must include the following two items:

- The application map that contains the application(s)
- A classifier



NOTE: You must also enable PFC on the code point of any traffic for which you want to achieve lossless transport.

Disabling DCBX Application Protocol Exchange

To disable DCBX application protocol exchange for all applications on an interface, issue the **set protocols dcbx interface *interface-name* applications no-auto-negotiation** command.

You can also disable DCBX application protocol exchange for applications on an interface by deleting the application map from the interface, or by deleting a particular application from the application map. However, when you delete an application from an application map, the application protocol is no longer exchanged on any interface which uses that application map.

On interfaces that use IEEE DCBX mode to exchange DCBX parameters, you can disable sending the enhanced transmission selection (ETS) Recommendation TLV to the peer if you want an asymmetric ETS configuration between the peers.

Related Documentation

- [Understanding DCBX on page 101](#)
- [Configuring DCBX Autonegotiation on page 112](#)
- [Disabling the ETS Recommendation TLV on page 115](#)
- [Defining an Application for DCBX Application Protocol TLV Exchange on page 120](#)
- [Configuring an Application Map for DCBX Application Protocol TLV Exchange on page 121](#)
- [Applying an Application Map to an Interface for DCBX Application Protocol TLV Exchange on page 122](#)
- [Example: Configuring DCBX Application Protocol TLV Exchange on page 123](#)

Defining an Application for DCBX Application Protocol TLV Exchange

Define each application for which you want DCBX to exchange application protocol information. You can define Layer 2 and Layer 4 applications. After you define applications, you map them to IEEE 802.1p code points, and then apply the application map to the interfaces on which you want DCBX to exchange application protocol information with connected peers. (See *Related Documentation* for how to configure application maps and apply them to interfaces, and for an example of the entire procedure that also includes classifier configuration.)



NOTE: In Junos OS Release 12.1, the FCoE application was configured by default, so you did not need to configure it in an application map. In Junos OS Release 12.2, if you want DCBX to advertise the FCoE application on an interface and you apply an application map to that interface, you must explicitly configure FCoE in the application map. You also must enable priority-based flow control (PFC) on the FCoE code point on all interfaces that you want to advertise FCoE. If you apply an application map to an interface, the interface sends DCBX TLVs only for the applications configured in the application map.

Define Layer 2 applications by mapping an application name to an EtherType. Define Layer 4 applications by mapping an application name to a protocol (TCP or UDP) and a destination port.

- To define a Layer 2 application, specify the name of the application and its EtherType:

```
[edit applications]
user@switch# set application application-name ether-type ether-type
```

For example, to configure an application named **PTP** (for Precision Time Protocol) that uses the EtherType **0x88F7**:

```
user@switch# set applications application ptp ether-type 0x88F7
```

- To define a Layer 4 application, specify the name of the application, its protocol (TCP or UDP), and its destination port:

```
[edit]
user@switch# set applications application application-name protocol (tcp | udp)
destination-port port-value
```

For example, to configure an application named **iscsi** (for Internet Small Computer System Interface) that uses the protocol **TCP** and the destination port **3260**:

```
user@switch# set applications application iscsi protocol tcp destination-port 3260
```

Related Documentation

- [Configuring an Application Map for DCBX Application Protocol TLV Exchange on page 121](#)
- [Applying an Application Map to an Interface for DCBX Application Protocol TLV Exchange on page 122](#)

- [Configuring DCBX Autonegotiation on page 112](#)
- [Example: Configuring DCBX Application Protocol TLV Exchange on page 123](#)
- [Example: Configuring DCBX to Support an iSCSI Application](#)
- [Understanding DCBX Application Protocol TLV Exchange on page 116](#)
- [show dcbx neighbors on page 169](#)

Configuring an Application Map for DCBX Application Protocol TLV Exchange

After you define applications for which you want to exchange DCBX application protocol information, map the applications to IEEE 802.1p code points. The IEEE 802.1p code points identify incoming traffic and allow you to map that traffic to the desired application. You then apply the application map to the interfaces on which you want DCBX to exchange application protocol information with connected peers. (See *Related Documentation* for how to define applications and apply the application map to interfaces, and for an example of the entire procedure that also includes classifier configuration.)



NOTE: In Junos OS Release 12.1, the FCoE application was configured by default, so you did not need to configure it in an application map. In Junos OS Release 12.2, if you want DCBX to advertise the FCoE application on an interface and you apply an application map to that interface, you must explicitly configure FCoE in the application map. You also must enable priority-based flow control (PFC) on the FCoE code point on all interfaces that you want to advertise FCoE. If you apply an application map to an interface, the interface sends DCBX TLVs only for the applications configured in the application map.

Configure an application map by creating an application map name and mapping an application to one or more IEEE 802.1p code points.

- To define an application map, specify the name of the application map, the name of the application, and the IEEE 802.1p code points of the incoming traffic that you want to associate with the application in the application map:

```
[edit policy-options]
user@switch# set application-maps application-map-name application application-name
code-points [ aliases ] [ bit-patterns ]
```

For example, to configure an application map named **ptp-app-map** that includes an application named **PTP** (for Precision Time Protocol) and map the application to IEEE 802.1p code points **001** and **101**:

```
user@switch# set policy-options application-maps ptp-app-map application ptp code points
[ 001 101 ]
```

Related Documentation

- [Defining an Application for DCBX Application Protocol TLV Exchange on page 120](#)
- [Applying an Application Map to an Interface for DCBX Application Protocol TLV Exchange on page 122](#)

- [Configuring DCBX Autonegotiation on page 112](#)
- [Example: Configuring DCBX Application Protocol TLV Exchange on page 123](#)
- [Example: Configuring DCBX to Support an iSCSI Application](#)
- [show dcbx neighbors on page 169](#)

Applying an Application Map to an Interface for DCBX Application Protocol TLV Exchange

After you define applications and map them to IEEE 802.1p code points in an application map, apply the application map to the interfaces on which you want DCBX to exchange the application protocol information with connected peers. (See *Related Documentation* for how to define applications and configure application maps to interfaces, and for an example of the entire procedure that also includes classifier configuration.)



NOTE: In Junos OS Release 12.1, the FCoE application was configured by default, so you did not need to configure it in an application map. In Junos OS Release 12.2, if you want DCBX to advertise the FCoE application on an interface and you apply an application map to that interface, you must explicitly configure FCoE in the application map. You also must enable priority-based flow control (PFC) on the FCoE code point on all interfaces that you want to advertise FCoE. If you apply an application map to an interface, the interface sends DCBX TLVs only for the applications configured in the application map.

- To apply an application map to a DCBX interface, specify the DCBX interface and the application map name:

[edit protocols]

```
user@switch# set dcbx interface interface-name application-map application-map-name
```

For example, to apply an application map named **ptp-app-map** on interface **xe-0/0/11**:

```
user@switch# set protocols dcbx interface xe-0/0/11 application-map ptp-app-map
```

Related Documentation

- [Defining an Application for DCBX Application Protocol TLV Exchange on page 120](#)
- [Configuring an Application Map for DCBX Application Protocol TLV Exchange on page 121](#)
- [Configuring DCBX Autonegotiation on page 112](#)
- [Example: Configuring DCBX Application Protocol TLV Exchange on page 123](#)
- [Example: Configuring DCBX to Support an iSCSI Application](#)
- [show dcbx neighbors on page 169](#)

Example: Configuring DCBX Application Protocol TLV Exchange

Data Center Bridging Capability Exchange protocol (DCBX) discovers the data center bridging (DCB) capabilities of connected peers by exchanging application configuration information. DCBX detects feature misconfiguration and mismatches and can configure DCB on peers. DCBX is an extension of the Link Layer Discovery Protocol (LLDP). LLDP must remain enabled on every interface on which you want to use DCBX.



NOTE: LLDP and DCBX are enabled by default on all interfaces.

The switch supports DCBX application protocol exchange for Layer 2 and Layer 4 applications such as the Internet Small Computer System Interface (iSCSI). You specify applications by EtherType (for Layer 2 applications) or by the destination port and protocol (for Layer 4 applications; the protocol can be either TCP or UDP).

The switch handles Fibre Channel over Ethernet (FCoE) application protocol exchange differently than other protocols in some cases:

- If FCoE is the only application for which you want to enable DCBX application protocol TLV exchange on an interface, you do not have to explicitly configure the FCoE application or an application map. By default, the switch exchanges FCoE application protocol TLVs on all interfaces that carry FCoE traffic (traffic mapped to the **fcoe** forwarding class) and have priority-based flow control (PFC) enabled on the FCoE priority (the FCoE IEEE 802.1p code point). The default priority mapping for the FCoE application is IEEE 802.1p code point 011 (the default **fcoe** forwarding class code point).
- If you want an interface to use DCBX to exchange application protocol TLVs for any other applications in addition to FCoE, you must configure the applications (including FCoE), define an application map (including FCoE), and apply the application map to the interface. If you apply an application map to an interface, you must explicitly configure the FCoE application, or the interface does not exchange FCoE application protocol TLVs.

This example shows how to configure interfaces to exchange both Layer 2 and Layer 4 applications by configuring one interface to exchange iSCSI and FCoE application protocol information and configuring another interface to exchange iSCSI and Precision Time Protocol (PTP) application protocol information.

- [Requirements on page 123](#)
- [Overview on page 124](#)
- [Configuration on page 127](#)
- [Verification on page 129](#)

Requirements

This example uses the following hardware and software components:

- Juniper Networks QFX Series device

- Junos OS Release 12.1 or later for the QFX Series

Overview

The switch supports DCBX application protocol exchange for:

- Layer 2 applications, defined by EtherType
- Layer 4 applications, defined by destination port and protocol



NOTE: DCBX also advertises PFC and enhanced transmission selection (ETS) information. See [“Configuring DCBX Autonegotiation” on page 112](#) for how DCBX negotiates and advertises configuration information for these features and for the applications.

DCBX is configured on a per-interface basis for each supported feature or application. For applications that you want to enable for DCBX application protocol exchange, you must:

- Define the application name and configure the EtherType or the destination port and protocol (TCP or UDP) of the application. Use the EtherType for Layer 2 applications, and use the destination port and protocol for Layer 4 protocols.
- Map the application to an IEEE 802.1p code point in an application map.
- Add the application map to DCBX interface.

In addition, for all applications (including FCoE, even when you do not use an application map), you either must create an IEEE 802.1p classifier and apply it to the appropriate ingress interfaces or use the default classifier. A classifier maps the code points of incoming traffic to a forwarding class and a loss priority so that ingress traffic is assigned to the correct class of service (CoS). The forwarding class determines the output queue on the egress interface.

If you do not create classifiers, trunk and tagged-access ports use the unicast IEEE 802.1 default trusted classifier. [Table 10 on page 124](#) shows the default mapping of IEEE 802.1 code-point values to unicast forwarding classes and loss priorities for ports in trunk mode or tagged-access mode. [Table 11 on page 125](#) shows the default untrusted classifier IEEE 802.1 code-point values to unicast forwarding class mapping for ports in access mode.

Table 10: Default IEEE 802.1 Classifiers for Trunk Ports and Tagged-Access Ports (Default Trusted Classifier)

Code Point	Forwarding Class	Loss Priority
be (000)	best-effort	low
be1 (001)	best-effort	low
ef (010)	best-effort	low

Table 10: Default IEEE 802.1 Classifiers for Trunk Ports and Tagged-Access Ports (Default Trusted Classifier) (continued)

Code Point	Forwarding Class	Loss Priority
ef1 (011)	fcoe	low
af11 (100)	no-loss	low
af12 (101)	best-effort	low
nc1 (110)	network-control	low
nc2 (111)	network-control	low

Table 11: Default IEEE 802.1 Unicast Classifiers for Access Ports (Default Untrusted Classifier)

Code Point	Forwarding Class	Loss Priority
000	best-effort	low
001	best-effort	low
010	best-effort	low
011	best-effort	low
100	best-effort	low
101	best-effort	low
110	best-effort	low
111	best-effort	low

Topology

This example shows how to configure DCBX application protocol exchange for three protocols (iSCSI, PTP, and FCoE) on two interfaces. One interface exchanges iSCSI and FCoE application protocol information, and the other interface exchanges iSCSI and PTP application protocol information.



NOTE: You must map FCoE traffic to the interfaces on which you want to forward FCoE traffic. You must also enable PFC on the FCoE interfaces and create an ingress classifier for FCoE traffic, or else use the default classifier.

Table 12 on page 126 shows the configuration components for this example.

Table 12: Components of DCBX Application Protocol Exchange Configuration Topology

Component	Settings
Hardware	QFX Series device
LLDP	Enabled by default on Ethernet interfaces
DCBX	Enabled by default on Ethernet interfaces
iSCSI application (Layer 4)	Application name— iscsi protocol— TCP destination-port— 3260 code-points— 111
PTP application (Layer 2)	Application name— ptp ether-type— 0x88F7 code-points— 001, 101
FCoE application (Layer 2)	Application name— fcoe ether-type— 0x8906 code-points— 011 NOTE: You explicitly configure the FCoE application because you are applying an application map to the interface. When you apply an application map to an interface, all applications must be explicitly configured and included in the application map.
Application maps	dcbx-iscsi-fcoe-app-map —Maps the iSCSI and FCoE applications to IEEE 802.1p code points dcbx-iscsi-ptp-app-map —Maps iSCSI and PTP applications to IEEE 802.1p code points
Interfaces	xe-0/0/10 —Configured to exchange FCoE and iSCSI application TLVs (uses application map dcbx-iscsi-fcoe-app-map , carries FCoE traffic, and has PFC enabled on the FCoE priority) xe-0/0/11 —Configured to exchange iSCSI and PTP application TLVs (uses application map dcbx-iscsi-ptp-app-map)
PFC congestion notification profile for FCoE application exchange	fcoe-cnp: <ul style="list-style-type: none"> Code point—011 Interface—xe-0/0/10

Table 12: Components of DCBX Application Protocol Exchange Configuration Topology (*continued*)

Component	Settings
Behavior aggregate classifiers (map forwarding classes to incoming packets by the packet's IEEE 802.1 code point)	<p>fcoe-iscsi-cl1:</p> <ul style="list-style-type: none"> Maps the fcoe forwarding class to the IEEE 802.1p code point used for the FCoE application (011) and a loss priority of high Maps the network-control forwarding class to the IEEE 802.1p code point used for the iSCSI application (111) and a loss priority of high Applied to interface xe-0/0/10 <p>iscsi-ntp-cl2:</p> <ul style="list-style-type: none"> Maps the network-control forwarding class to the IEEE 802.1p code point used for the iSCSI application (111) and a loss priority of low Maps the best-effort forwarding class to the IEEE 802.1p code points used for the PTP application (001 and 101) and a loss priority of low Applied to interface xe-0/0/11



NOTE: This example does not include scheduling (bandwidth allocation) configuration or lossless configuration for the iSCSI forwarding class.

Configuration

CLI Quick Configuration

To quickly configure DCBX application protocol exchange, copy the following commands, paste them in a text file, remove line breaks, change variables and details to match your network configuration, and then copy and paste the commands into the CLI at the **[edit]** hierarchy level.

```
set applications application iSCSI protocol tcp destination-port 3260
set applications application FCoE ether-type 0x8906
set applications application PTP ether-type 0x88F7
set policy-options application-maps dcbx-iscsi-fcoe-app-map application iSCSI code-points 111
set policy-options application-maps dcbx-iscsi-fcoe-app-map application FCoE code-points 011
set policy-options application-maps dcbx-iscsi-ntp-app-map application iSCSI code-points 111
set policy-options application-maps dcbx-iscsi-ntp-app-map application PTP code-points [001 101]
set protocols dcbx interface xe-0/0/10 application-map dcbx-iscsi-fcoe-app-map
set protocols dcbx interface xe-0/0/11 application-map dcbx-iscsi-ntp-app-map
set class-of-service congestion-notification-profile fcoe-cnp input ieee-802.1 code-point 011 pfc
set class-of-service interfaces xe-0/0/10 congestion-notification-profile fcoe-cnp
set class-of-service classifiers ieee-802.1 fcoe-iscsi-cl1 import default forwarding-class fcoe
loss-priority high code-points 011
set class-of-service classifiers ieee-802.1 fcoe-iscsi-cl1 import default forwarding-class
network-control loss-priority high code-points 111
set class-of-service classifiers ieee-802.1 iscsi-ntp-cl2 import default forwarding-class
network-control loss-priority low code-points 111
set class-of-service classifiers ieee-802.1 iscsi-ntp-cl2 import default forwarding-class best-effort
loss-priority low code-points [001 101]
set class-of-service interfaces xe-0/0/10 unit 0 classifiers ieee-802.1 fcoe-iscsi-cl1
```

```
set class-of-service interfaces xe-0/0/11 unit 0 classifiers ieee-802.1 iscsi-ptp-cl2
```

Configuring DCBX Application Protocol TLV Exchange

Step-by-Step Procedure

To define the applications, map the applications to IEEE 802.1p code points, apply the applications to interfaces, and create classifiers for DCBX application protocol exchange:

1. Define the iSCSI application by specifying its protocol and destination port, and define the FCoE and PTP applications by specifying their EtherTypes.

```
[edit applications]
user@switch# set application iSCSI protocol tcp destination-port 3260
user@switch# set application FCoE ether-type 0x8906
user@switch# set application PTP ether-type 0x88F7
```

2. Define an application map that maps the iSCSI and FCoE applications to IEEE 802.1p code points.

```
[edit policy-options]
user@switch# set application-maps dcbx-iscsi-fcoe-app-map application iSCSI code-points 111
user@switch# set application-maps dcbx-iscsi-fcoe-app-map application FCoE code-points 011
```

3. Define the application map that maps the iSCSI and PTP applications to IEEE 802.1p code points.

```
[edit policy-options]
user@switch# set application-maps dcbx-iscsi-ptp-app-map application iSCSI code-points 111
user@switch# set application-maps dcbx-iscsi-ptp-app-map application PTP code-points [001 101]
```

4. Apply the iSCSI and FCoE application map to interface **xe-0/0/10**, and apply the iSCSI and PTP application map to interface **xe-0/0/11**.

```
[edit protocols dcbx]
user@switch# set interface xe-0/0/10 application-map dcbx-iscsi-fcoe-app-map
user@switch# set interface xe-0/0/11 application-map dcbx-iscsi-ptp-app-map
```

5. Create the congestion notification profile to enable PFC on the FCoE code point (011), and apply the congestion notification profile to interface **xe-0/0/10**.

```
[edit class-of-service]
user@switch# set congestion-notification-profile fcoe-cnp input ieee-802.1 code-point 011 pfc
user@switch# set interfaces xe-0/0/10 congestion-notification-profile fcoe-cnp
```

6. Configure the classifier to apply to the interface that exchanges iSCSI and FCoE application information.

```
[edit class-of-service classifiers]
user@switch# set ieee-802.1 fcoe-iscsi-cl1 import default forwarding-class fcoe loss-priority high code-points 011
user@switch# set ieee-802.1 fcoe-iscsi-cl1 import default forwarding-class network-control loss-priority high code-points 111
```

7. Configure the classifier to apply to the interface that exchanges iSCSI and PTP application information.

```
[edit class-of-service classifiers]
```



```

user@switch# set ieee-802.1 iscsi-ptp-cl2 import default forwarding-class network-control
loss-priority low code-points 111
user@switch# set ieee-802.1 iscsi-ptp-cl2 import default forwarding-class best-effort
loss-priority low code-points [001 101]

```

8. Apply the classifiers to the appropriate interfaces.

```

[edit class-of-service]
user@switch# set interfaces xe-0/0/10 unit 0 classifiers ieee-802.1 fcoe-iscsi-cl1
user@switch# set interfaces xe-0/0/11 unit 0 classifiers ieee-802.1 iscsi-ptp-cl2

```

Verification

To verify that DCBX application protocol exchange configuration has been created and is operating properly, perform these tasks:

- [Verifying the Application Configuration on page 129](#)
- [Verifying the Application Map Configuration on page 129](#)
- [Verifying DCBX Application Protocol Exchange Interface Configuration on page 130](#)
- [Verifying the PFC Configuration on page 130](#)
- [Verifying the Classifier Configuration on page 131](#)

Verifying the Application Configuration

Purpose Verify that DCBX applications have been configured.

Action List the applications by using the configuration mode command **show applications**:

```

user@switch# show applications
application iSCSI {
    protocol tcp;
    destination-port 3260;
}

application fcoe {
    ether-type 0x8906;
}

application ptp {
    ether-type 0x88F7;
}

```

Meaning The **show applications** configuration mode command lists all of the configured applications and either their protocol and destination port (Layer 4 applications) or their EtherType (Layer 2 applications). The command output shows that the iSCSI application is configured with the **tcp** protocol and destination port **3260**, the FCoE application is configured with the EtherType **0x8906**, and that the PTP application is configured with the EtherType **0x88F7**.

Verifying the Application Map Configuration

Purpose Verify that the application maps have been configured.

Action List the application maps by using the configuration mode command **show policy-options application-maps**:

```
user@switch# show policy-options application-maps
dcbx-iscsi-fcoe-app-map {
    application iSCSI code-points 111;
    application FCoE code-points 011;
}

dcbx-iscsi-ptp-app-map {
    application iSCSI code-points 111;
    application PTP code-points [001 101];
}
```

Meaning The **show policy-options application-maps** configuration mode command lists all of the configured application maps and the applications that belong to each application map. The command output shows that there are two application maps, **dcbx-iscsi-fcoe-app-map** and **dcbx-iscsi-ptp-app-map**.

The application map **dcbx-iscsi-fcoe-app-map** consists of the iSCSI application, which is mapped to IEEE 802.1p code point 111, and the FCoE application, which is mapped to IEEE 802.1p code point 011.

The application map **dcbx-iscsi-ptp-app-map** consists of the iSCSI application, which is mapped to IEEE 802.1p code point 111, and the PTP application, which is mapped to IEEE 802.1p code points 001 and 101.

Verifying DCBX Application Protocol Exchange Interface Configuration

Purpose Verify that the application maps have been applied to the correct interfaces.

Action List the application maps by using the configuration mode command **show protocols dcbx**:

```
user@switch# show protocols dcbx
interface xe-0/0/10.0 {
    application-map dcbx-iscsi-fcoe-app-map;
}

interface xe-0/0/11.0 {
    application-map dcbx-iscsi-ptp-app-map;
}
```

Meaning The **show protocols dcbx** configuration mode command lists whether the interfaces are enabled for DCBX and lists the application map applied to each interface. The command output shows that interfaces **xe-0/0/10.0** and **xe-0/0/11.0** are enabled for DCBX, and that interface **xe-0/0/10.0** uses application map **dcbx-iscsi-fcoe-app-map**, and interface **xe-0/0/11.0** uses application map **dcbx-iscsi-ptp-app-map**.

Verifying the PFC Configuration

Purpose Verify that PFC has been enabled on the FCoE code point and applied to the correct interface.

Action Display the PFC configuration to verify that PFC is enabled on the FCoE code point (011) in the congestion notification profile **fcoe-cnp** by using the configuration mode command **show class-of-service congestion-notification-profile**:

```
user@switch# show class-of-service congestion-notification-profile
fcoe-cnp {
  input {
    ieee-802.1 {
      code-point 011 {
        pfc;
      }
    }
  }
}
```

Display the class-of-service (CoS) interface information to verify that the correct interface has PFC enabled for the FCoE application by using the configuration mode command **show class-of-service interfaces**:

```
user@switch# show class-of-service interfaces
xe-0/0/10 {
  congestion-notification-profile fcoe-cnp;
}
```



NOTE: The sample output does not include all of the information this command can show. The output is abbreviated to focus on verifying the PFC configuration.

Meaning The **show class-of-service congestion-notification-profile** configuration mode command lists the configured congestion notification profiles. The command output shows that the congestion notification profile **fcoe-cnp** has been configured and has enabled PFC on the IEEE 802.1p code point **011** (the default FCoE code point).

The **show class-of-service interfaces** configuration mode command shows the interface CoS configuration. The command output shows that the congestion notification profile **fcoe-cnp**, which enables PFC on the FCoE code point, is applied to interface **xe-0/0/10**.

Verifying the Classifier Configuration

Purpose Verify that the classifiers have been configured and applied to the correct interfaces.

Action Display the classifier configuration by using the configuration mode command **show class-of-service**:

```
user@switch# show class-of-service
classifiers {
  ieee-802.1 fcoe-iscsi-cl1 {
    import default;
    forwarding-class network-control {
      loss-priority high code-points 111;
    }
    forwarding-class fcoe {
```

```

        loss-priority high code-points 011;
    }
}
ieee-802.1 iscsi-ntp-cl2 {
    import default;
    forwarding-class network-control {
        loss-priority low code-points 111;
    }
    forwarding-class best-effort {
        loss-priority low code-points [ 001 101 ];
    }
}
}
interfaces {
    xe-0/0/10 {
        congestion-notification-profile fcoe-cnp;
        unit 0 {
            classifiers {
                ieee-802.1 fcoe-iscsi-cl1;
            }
        }
    }
    xe-0/0/11 {
        unit 0 {
            classifiers {
                ieee-802.1 iscsi-ntp-cl2;
            }
        }
    }
}
}

```



NOTE: The sample output does not include all of the information this command can show. The output is abbreviated to focus on verifying the classifier configuration.

Meaning The **show class-of-service** configuration mode command lists the classifier and CoS interface configuration, as well as other information not shown in this example. The command output shows that there are two classifiers configured, **fcoe-iscsi-cl1** and **iscsi-ntp-cl2**.

Classifier **fcoe-iscsi-cl1** uses the **default** classifier as a template and edits the template as follows:

- The forwarding class **network-control** is set to a loss priority of **high** and is mapped to code point **111** (the code point mapped to the iSCSI application).
- The forwarding class **fcoe** is set to a loss priority of **high** and is mapped to code point **011** (the code point mapped by default to the FCoE application).

Classifier **iscsi-ntp-cl2** uses the **default** classifier as a template and edits the template as follows:

- The forwarding class **network-control** is set to a loss priority of **low** and is mapped to IEEE 802.1p code point **111** (the code point mapped to the iSCSI application).
- The forwarding class **best-effort** is set to a loss priority of **low** and is mapped to IEEE 802.1p code points **001** and **101** (the code points mapped by default to the PTP application).

The command output also shows that classifier **fcoe-iscsi-cl1** is mapped to interface **xe-0/0/10.0** and that classifier **iscsi-ntp-cl2** is mapped to interface **xe-0/0/11.0**.

Related Documentation

- [Defining an Application for DCBX Application Protocol TLV Exchange on page 120](#)
- [Configuring an Application Map for DCBX Application Protocol TLV Exchange on page 121](#)
- [Applying an Application Map to an Interface for DCBX Application Protocol TLV Exchange on page 122](#)
- [Configuring DCBX Autonegotiation on page 112](#)
- [show dcbx on page 168](#)
- [show dcbx neighbors on page 169](#)
- [Understanding DCBX Application Protocol TLV Exchange on page 116](#)

CHAPTER 4

Learn About Technology

- [Data Center Technology Overview Videos on page 135](#)

Data Center Technology Overview Videos

Juniper Information Experience (IX) videos provide brief, high-level overviews of data center technologies and concepts. Each video runs approximately one-and-a-half to two minutes in length. This document contains SDN-related videos and links to conceptual documents that contain other data center technology videos:

- [Learn About Video: Why Do We Need an IP Fabric? on page 135](#)
- [Learn About Video: What is the Best Control Plane Protocol to Use in a Data Center IP Fabric? on page 135](#)
- [Learn About Video: Why Use an Overlay Network in a Data Center? on page 135](#)
- [Conceptual Documents That Contain Technology Overview Videos on page 136](#)

Learn About Video: Why Do We Need an IP Fabric?

The video *Why Do We Need an IP Fabric?* presents a brief overview of IP Fabric use cases.



Video: [Why Do We Need an IP Fabric?](#)

Learn About Video: What is the Best Control Plane Protocol to Use in a Data Center IP Fabric?

The video *What is the Best Control Plane Protocol to Use in a Data Center IP Fabric?* presents a brief overview of the arguments for using Border Gateway Protocol (BGP) as the data center IP fabric control plane protocol.



Video: [What is the Best Control Plane Protocol to Use in a Data Center IP Fabric?](#)

Learn About Video: Why Use an Overlay Network in a Data Center?

The video *Why Use an Overlay Network in a Data Center?* presents a brief overview of the advantages of data center overlay networks.



Video: [Why Use an Overlay Network in a Data Center?](#)

Conceptual Documents That Contain Technology Overview Videos

The following conceptual documents include brief video overviews of the technology:

- [Understanding DCB Features and Requirements on page 78](#)
- *Understanding CoS Hierarchical Port Scheduling (ETS)*
- [Understanding CoS Flow Control \(Ethernet PAUSE and PFC\) on page 81](#)
- [Understanding DCBX on page 101](#)
- *Understanding PFC Functionality Across Layer 3 Interfaces*
- *Virtual Chassis Fabric Overview*
- *Understanding In-Service Software Upgrade (ISSU) and In-Service Software Upgrade (ISSU) System Requirements (same video)*

PART 3

Configuration Statements and Operational Commands

- [Configuration Statements \(FCoE on a Transit Switch\) on page 139](#)
- [Configuration Statements \(DCBX and PFC\) on page 143](#)
- [Operational Commands \(FCoE on a Transit Switch\) on page 163](#)
- [Operational Commands \(DCBX and PFC\) on page 167](#)

CHAPTER 5

Configuration Statements (FCoE on a Transit Switch)

- [family fcoe on page 140](#)
- [oxid on page 141](#)

family fcoe

Syntax	<p>QFX Series Standalone Switches</p> <pre>family fcoe { oxid (enable disable); }</pre> <p>QFabric Systems</p> <pre>family fcoe { ethernet-interfaces { node-group (node-group-name all) { oxid (enable disable); } } fabric-interfaces { node-group (node-group-name all) { oxid (enable disable); } } }</pre>
Hierarchy Level	[edit forwarding-options hash-key]
Release Information	<p>Statement introduced in Junos OS Release 12.3 for the QFX Series.</p> <p>Ethernet-interfaces and fabric-interfaces statements introduced in Junos OS Release 13.2X52-D10 for the QFX Series.</p>
Description	Configure whether or not to use the originator exchange identifier (Oxid) field for hash control for FCoE traffic load balancing.
Options	The statement is explained separately.
Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • Enabling and Disabling CoS OxID Hash Control on Standalone Switches on page 73 • Understanding OxID Hash Control for FCoE Traffic Load Balancing on Standalone Switches on page 72

oxid

Syntax	oxid (enable disable)
Hierarchy Level	<p>QFX Series Standalone Switches</p> <p>[edit forwarding-options hash-key family fcoe]</p> <p>QFabric Systems</p> <p>[edit forwarding-options hash-key family fcoe ethernet-interfaces node-group (node-group-name all) {}]</p> <p>[edit forwarding-options hash-key family fcoe fabric-interfaces node-group (node-group-name all) {}]</p>
Release Information	<p>Statement introduced in Junos OS Release 12.3 for the QFX Series.</p> <p>Statement introduced in Junos OS Release 13.2X52-D10 for the QFabric System.</p>
Description	Enable or disable whether the switch uses the originator exchange identifier (OxID) field for hash control for FCoE traffic load balancing.
Default	OxID hash control is enabled by default.
Options	oxid (enable disable) —Enable or disable whether the switch uses the OxID hash control field for FCoE traffic load balancing.
Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • Enabling and Disabling CoS OxID Hash Control on Standalone Switches on page 73 • Understanding OxID Hash Control for FCoE Traffic Load Balancing on Standalone Switches on page 72

CHAPTER 6

Configuration Statements (DCBX and PFC)

- [application \(Application Maps\) on page 144](#)
- [application \(Applications\) on page 145](#)
- [application-map on page 146](#)
- [application-maps on page 147](#)
- [applications \(Applications\) on page 148](#)
- [applications \(DCBX\) on page 149](#)
- [code-points \(Application Maps\) on page 150](#)
- [dcbx on page 151](#)
- [dcbx-version on page 152](#)
- [destination-port \(Applications\) on page 153](#)
- [disable \(DCBX\) on page 154](#)
- [enhanced-transmission-selection on page 155](#)
- [ether-type on page 156](#)
- [interface \(DCBX\) on page 157](#)
- [no-recommendation-tlv on page 158](#)
- [policy-options on page 159](#)
- [priority-flow-control on page 160](#)
- [protocol \(Applications\) on page 161](#)
- [recommendation-tlv on page 162](#)

application (Application Maps)

Syntax	<code>application <i>application-name</i> { <i>code-points</i> [<i>aliases</i>] [<i>bit-patterns</i>]; }</code>
Hierarchy Level	[edit policy-options application-maps <i>application-map-name</i>]
Release Information	Statement introduced in Junos OS Release 12.1 for EX Series switches. Statement introduced in Junos OS Release 12.1 for the QFX Series.
Description	Add an application to an application map and define the application's code points.
Options	<i>application-name</i> —Name of the application. The remaining statement is explained separately.
Required Privilege Level	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Configuring an Application Map for DCBX Application Protocol TLV Exchange on page 121• Example: Configuring DCBX Application Protocol TLV Exchange on page 123• Example: Configuring DCBX to Support an iSCSI Application• Understanding DCBX Application Protocol TLV Exchange on page 116• Understanding DCBX Application Protocol TLV Exchange on EX Series Switches

application (Applications)

Syntax	<pre> application <i>application-name</i> { <i>destination-port</i> <i>port-value</i>; <i>protocol</i> (tcp udp); <i>ether-type</i> <i>type</i>; } </pre>
Hierarchy Level	[edit applications]
Release Information	Statement introduced in Junos OS Release 12.1 for EX Series switches. Statement introduced in Junos OS Release 12.1 for the QFX Series.
Description	Configure properties to define an application.
Options	<p><i>application-name</i>—Name of the application.</p> <p>The statements are explained separately.</p>
Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • Defining an Application for DCBX Application Protocol TLV Exchange on page 120 • Example: Configuring DCBX Application Protocol TLV Exchange on page 123 • Example: Configuring DCBX to Support an iSCSI Application • Understanding DCBX Application Protocol TLV Exchange on page 116 • Understanding DCBX Application Protocol TLV Exchange on EX Series Switches

application-map

Syntax	<code>application-map <i>application-map-name</i>;</code>
Hierarchy Level	[edit protocols dcbx interface <i>interface-name</i>]
Release Information	Statement introduced in Junos OS Release 12.1 for EX Series switches. Statement introduced in Junos OS Release 12.1 for the QFX Series.
Description	Specify an application map to apply to an interface.
Options	<i>application-map-name</i> —Name of the application map.
Required Privilege Level	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• show dcbx neighbors on page 169• Applying an Application Map to an Interface for DCBX Application Protocol TLV Exchange on page 122• Example: Configuring DCBX Application Protocol TLV Exchange on page 123• Example: Configuring DCBX to Support an iSCSI Application• Understanding DCBX Application Protocol TLV Exchange on page 116• Understanding DCBX Application Protocol TLV Exchange on EX Series Switches

application-maps

Syntax	<pre> application-maps <i>application-map-name</i> { application <i>application-name</i> { code-points [<i>aliases</i>] [<i>bit-patterns</i>]; } } </pre>
Hierarchy Level	[edit policy-options]
Release Information	<p>Statement introduced in Junos OS Release 12.1 for EX Series switches.</p> <p>Statement introduced in Junos OS Release 12.1 for the QFX Series.</p>
Description	Define an application map by specifying the applications that belong to the application map.
Options	<p><i>application-map-name</i>—Name of the application map.</p> <p>The remaining statements are explained separately.</p>
Required Privilege Level	<p>routing—To view this statement in the configuration.</p> <p>routing-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • Configuring an Application Map for DCBX Application Protocol TLV Exchange on page 121 • Example: Configuring DCBX Application Protocol TLV Exchange on page 123 • Example: Configuring DCBX to Support an iSCSI Application • Understanding DCBX Application Protocol TLV Exchange on page 116 • Understanding DCBX Application Protocol TLV Exchange on EX Series Switches

applications (Applications)

Syntax	<pre>applications { application application-name { destination-port port-value; protocol (tcp udp); ether-type type; } }</pre>
Hierarchy Level	[edit]
Release Information	Statement introduced in Junos OS Release 12.1 for EX Series switches. Statement introduced in Junos OS Release 12.1 for the QFX Series.
Description	Define applications that DCBX advertises.
Options	The statements are explained separately.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Defining an Application for DCBX Application Protocol TLV Exchange on page 120• Example: Configuring DCBX Application Protocol TLV Exchange on page 123• Example: Configuring DCBX to Support an iSCSI Application• Understanding DCBX Application Protocol TLV Exchange on page 116• Understanding DCBX Application Protocol TLV Exchange on EX Series Switches

applications (DCBX)

Syntax	<pre>applications { fcoe { no-auto-negotiation; } }</pre>
Hierarchy Level	[edit protocols dcbx interface <i>interface-name</i>]
Release Information	Statement introduced in Junos OS Release 11.1 for the QFX Series. Statement introduced in Junos OS Release 12.1 for the EX Series
Description	Configure Data Center Bridging Capability Exchange protocol (DCBX) applications on an interface.
Options	The remaining statements are explained separately.
Required Privilege Level	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• show dcbx neighbors on page 169• Understanding DCB Features and Requirements on page 78

code-points (Application Maps)

Syntax	<code>code-points [<i>aliases</i>] [<i>bit-patterns</i>];</code>
Hierarchy Level	[edit policy-options application-maps <i>application-map-name</i> application <i>application-name</i>]
Release Information	Statement introduced in Junos OS Release 12.1 for EX Series switches. Statement introduced in Junos OS Release 12.1 for the QFX Series.
Description	Define one or more code-point aliases or bit sets for an application.
Options	<i>aliases</i> —Name of the alias or aliases. <i>bit-patterns</i> —Value of the code-point bits, in decimal form.
Required Privilege Level	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Configuring an Application Map for DCBX Application Protocol TLV Exchange on page 121• Example: Configuring DCBX Application Protocol TLV Exchange on page 123• Example: Configuring DCBX to Support an iSCSI Application• Understanding DCBX Application Protocol TLV Exchange on page 116• Understanding DCBX Application Protocol TLV Exchange on EX Series Switches


dcbx

Syntax	<pre> dcbx { disable; interface (interface-name all) { disable; application-map application-map-name; applications { fcoe { no-auto-negotiation; } } enhanced-transmission-selection { no-auto-negotiation; no-recommendation-tlv; recommendation-tlv { no-auto-negotiation; } } dcbx-version (auto-negotiate ieee-dcbx dcbx-version-1.01); priority-flow-control { no-auto-negotiation; } } } </pre>
Hierarchy Level	[edit protocols]
Release Information	<p>Statement introduced in Junos OS Release 11.1 for the QFX Series.</p> <p>Statement introduced in Junos OS Release 11.3 for EX Series switches.</p> <p>mode and recommendation-tlv statements introduced in Junos OS Release 12.2 for the QFX Series.</p>
Description	Configure DCBX properties.
Options	The statements are explained separately.
Required Privilege Level	<p>routing—To view this statement in the configuration.</p> <p>routing-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • show dcbx neighbors on page 169 • Understanding DCB Features and Requirements on page 78 • Configuring DCBX Autonegotiation on page 112 • <i>Understanding DCB Features and Requirements on EX Series Switches</i> • <i>Disabling DCBX to Disable PFC Autonegotiation on EX Series Switches (CLI Procedure)</i>

dcbx-version

Syntax	<code>dcbx-version (auto-negotiate ieee-dcbx dcbx-version-1.01);</code>
Hierarchy Level	[edit protocols dcbx interface (all <i>interface-name</i>)]
Release Information	Statement introduced in Junos OS Release 12.2 for the QFX Series.
Description	<p>Set the DCBX version for the specified interface or interfaces.</p> <p>QFX3500 switches come up in IEEE DCBX mode and then autonegotiate with the connected peer to set the DCBX version.</p> <p>QFabric system Node devices come up using DCBX version 1.01, and then autonegotiate with the connected peer to set the DCBX mode.</p>
Default	The default DCBX mode is autonegotiation.
Options	<p>auto-negotiate—Automatically negotiate the DCBX version with the connected peer.</p> <p>ieee-dcbx—Force the interface to use IEEE DCBX mode, regardless of the peer configuration.</p> <p>dcbx-version-1.01—Force the interface to use version 1.01 DCBX mode, regardless of the peer configuration.</p>
Required Privilege Level	<p>routing—To view this statement in the configuration.</p> <p>routing-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none">• show dcbx neighbors on page 169• Configuring DCBX Autonegotiation on page 112• Understanding DCBX on page 101

destination-port (Applications)

Syntax	<code>destination-port <i>port-value</i>;</code>
Hierarchy Level	[edit applications application <i>application-name</i>]
Release Information	Statement introduced in Junos OS Release 12.1 for EX Series switches. Statement introduced in Junos OS Release 12.1 for the QFX Series.
Description	Transmission Control Protocol (TCP) or User Datagram Protocol (UDP) destination port number, which combines with protocol to identify an application type. The Internet Assigned Numbers Authority (IANA) assigns port numbers. See the IANA <i>Service Name and Transport Protocol Port Number Registry</i> at http://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xml for a list of assigned port numbers.
<div>  NOTE: To create an application for iSCSI, use the protocol <code>tcp</code> with the destination port number <code>3260</code>. </div>	
Options	<i>port-value</i> —Identifier for the port.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • Defining an Application for DCBX Application Protocol TLV Exchange on page 120 • Example: Configuring DCBX Application Protocol TLV Exchange on page 123 • Example: Configuring DCBX to Support an iSCSI Application • Understanding DCBX Application Protocol TLV Exchange on page 116 • Understanding DCBX Application Protocol TLV Exchange on EX Series Switches


disable (DCBX)

Syntax	disable
Hierarchy Level	[edit protocols dcbx] [edit protocols dcbx interface <i>interface-name</i>]
Release Information	Statement introduced in Junos OS Release 11.1 for the QFX Series. Statement introduced in Junos OS Release 11.3 for EX Series switches.
Description	Disable Data Center Bridging Capability Exchange protocol (DCBX) on one or more 10-Gigabit Ethernet interfaces.
Default	DCBX is enabled by default on all 10-Gigabit or higher Ethernet interfaces. DCBX is enabled by default on all 10-Gigabit Ethernet interfaces on EX4500 CEE-enabled switches.
Required Privilege Level	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Configuring DCBX Autonegotiation on page 112• <i>Disabling DCBX to Disable PFC Autonegotiation on EX Series Switches (CLI Procedure)</i>• Understanding DCB Features and Requirements on page 78• <i>Understanding DCB Features and Requirements on EX Series Switches</i>

enhanced-transmission-selection

Syntax	<pre> enhanced-transmission-selection { no-auto-negotiation; no-recommendation-tlv; recommendation-tlv { no-auto-negotiation; } } </pre>
Hierarchy Level	[edit protocols dcbx interface <i>interface-name</i>]
Release Information	Statement introduced in Junos OS Release 11.1 for the QFX Series.
Description	<p>Disable advertising the enhanced transmission selection (ETS) state of the interface to the peer. To disable ETS on the interface, do not enable ETS on the interface in the class-of-service (CoS) configuration.</p> <p>Disabling ETS autonegotiation stops the QFX Series from advertising the ETS Configuration TLV and the ETS Recommendation TLV.</p> <p>Disabling the ETS recommendation TLV stops the QFX Series from advertising the ETS Recommendation TLV, but the ETS Configuration TLV is still advertised.</p>
Options	<p>no-auto-negotiation—Disable automatic negotiation of ETS (Configuration TLV and Recommendation TLV)</p> <p>no-recommendation-tlv—Disable automatic negotiation of the ETS Recommendation TLV</p> <p>recommendation-tlv—Enable automatic negotiation of ETS Recommendation TLV</p>
Required Privilege Level	<p>routing—To view this statement in the configuration.</p> <p>routing-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • show dcbx neighbors on page 169 • Configuring DCBX Autonegotiation on page 112 • Example: Configuring CoS Hierarchical Port Scheduling (ETS) • Understanding DCB Features and Requirements on page 78

ether-type

Syntax	<code>ether-type <i>ether-type</i>;</code>
Hierarchy Level	[edit applications application <i>application-name</i>]
Release Information	Statement introduced in Junos OS Release 12.1 for EX Series switches. Statement introduced in Junos OS Release 12.1 for the QFX Series.
Description	Two-octet field in an Ethernet frame that defines the protocol encapsulated in the frame payload. See http://standards.ieee.org/develop/regauth/ethertype/eth.txt for a list of Institute of Electrical and Electronics Engineers (IEEE) EtherTypes.
<div> NOTE: To create a FIP application, use the EtherType 0x8914.</div>	
Options	type —Identifier for the EtherType.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Defining an Application for DCBX Application Protocol TLV Exchange on page 120• Example: Configuring DCBX Application Protocol TLV Exchange on page 123• Understanding DCBX Application Protocol TLV Exchange on page 116

interface (DCBX)

Syntax	<pre> interface (<i>interface-name</i> all) { disable; application-map <i>application-map-name</i>; applications { no-auto-negotiation; } enhanced-transmission-selection { no-auto-negotiation; no-recommendation-tlv; recommendation-tlv { no-auto-negotiation; } } dcbx-version (auto-negotiate ieee-dcbx dcbx-version-1.01); priority-flow-control { no-auto-negotiation; } } </pre>
Hierarchy Level	[edit protocols dcbx]
Release Information	<p>Statement introduced in Junos OS Release 11.1 for the QFX Series.</p> <p>Statement introduced in Junos OS Release 11.3 for the EX Series switches.</p> <p>Mode and recommendation-tlv statements introduced in Junos OS Release 12.2 for the QFX Series.</p>
Description	Configure DCBX properties on an interface.
Options	<p><i>interface-name</i>—Name of the interface.</p> <p>The remaining statements are explained separately.</p>
Required Privilege Level	<p>routing—To view this statement in the configuration.</p> <p>routing-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • show dcbx neighbors on page 169 • Configuring DCBX Autonegotiation on page 112 • Example: Configuring DCBX to Support an iSCSI Application • Understanding DCB Features and Requirements on page 78 • Understanding DCB Features and Requirements on EX Series Switches • Understanding DCBX Application Protocol TLV Exchange on EX Series Switches

no-recommendation-tlv

Syntax	no-recommendation-tlv;
Hierarchy Level	[edit protocols dcbx interface <i>interface-name</i> enhanced-transmission-selection]
Release Information	Statement introduced in Junos OS Release 12.2 for the QFX Series.
Description	Disable DCBX to send the ETS Recommendation TLV (also known as the Information TLV) on egress. This feature is valid only if the interface DCBX mode is IEEE DCBX. If the interface DCBX mode is DCBX version 1.01, this statement has no effect. (DCBX version 1.01 does not advertise separate TLVs for individual attributes.)
Default	DCBX-enabled interfaces send the ETS recommendation TLV unless it is disabled.
Required Privilege Level	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• show dcbx neighbors on page 169• Configuring DCBX Autonegotiation on page 112

policy-options

```
Syntax  policy-options
        application-maps application-map-name {
            application application-name {
                code-points [ aliases ] [ bit-patterns ];
            }
        }
        policy-statement policy-name {
            term term-name {
                from {
                    family family-name;
                    match-conditions;
                    policy subroutine-policy-name;
                    prefix-list prefix-list-name;
                    prefix-list-filter prefix-list-name match-type <actions>;
                    route-filter destination-prefix match-type <actions>;
                    source-address-filter source-prefix match-type <actions>;
                }
                to {
                    match-conditions;
                    policy subroutine-policy-name;
                }
                then actions;
            }
        }
    }
```

Hierarchy Level [edit]

Release Information Statement introduced in Junos OS Release 12.1 for the QFX Series.
Statement introduced in Junos OS Release 12.1 for the EX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description Configure options such as application maps for DCBX application protocol exchange and policy statements.

Required Privilege Level storage—To view this statement in the configuration.
storage-control—To add this statement to the configuration.


Related Documentation

- [Defining an Application for DCBX Application Protocol TLV Exchange on page 120](#)
- [Example: Configuring DCBX Application Protocol TLV Exchange on page 123](#)
- [Example: Configuring DCBX to Support an iSCSI Application](#)
- [Understanding DCBX Application Protocol TLV Exchange on page 116](#)
- [Understanding DCBX Application Protocol TLV Exchange on EX Series Switches](#)

priority-flow-control

Syntax	<pre>priority-flow-control { no-auto-negotiation; }</pre>
Hierarchy Level	[edit protocols dcbx interface (all <i>interface-name</i>)]
Release Information	Statement introduced in Junos OS Release 11.1 for the QFX Series. Statement introduced in Junos OS Release 11.3 for EX Series switches.
Description	Disable autonegotiation of priority-based flow control (PFC) on one or more Ethernet interfaces. Autonegotiation enables PFC on an interface only if the switch and the peer device connected to the switch both support PFC and have the same PFC configuration. Disabling autonegotiation on an interface forces the interface to use the PFC state (enabled or disabled) that is configured on the switch by the configuration and assignment of the congestion notification profile.
Options	no-auto-negotiation —Disable automatic negotiation of PFC.
Required Privilege Level	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• show dcbx neighbors on page 169• <i>Configuring CoS PFC (Congestion Notification Profiles)</i>• <i>Configuring Priority-Based Flow Control for an EX Series Switch (CLI Procedure)</i>• Configuring DCBX Autonegotiation on page 112• Example: Configuring CoS PFC for FCoE Traffic on page 92• <i>Understanding Data Center Bridging Capability Exchange Protocol for EX Series Switches</i>• <i>Understanding Priority-Based Flow Control</i>• Understanding DCB Features and Requirements on page 78

protocol (Applications)

Syntax	<code>protocol (tcp udp);</code>
Hierarchy Level	[edit applications application <i>application-name</i>]
Release Information	Statement introduced in Junos OS Release 12.1 for EX Series switches. Statement introduced in Junos OS Release 12.1 for the QFX Series.
Description	Networking protocol type, which combines with destination-port to identify an application type.
<div>  NOTE: To create an application for iSCSI, use the protocol tcp with the destination port number 3260. </div>	
Options	tcp —Transmission Control Protocol udp —User Datagram Protocol
Required Privilege Level	interface —To view this statement in the configuration. interface-control —To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • Defining an Application for DCBX Application Protocol TLV Exchange on page 120 • Example: Configuring DCBX Application Protocol TLV Exchange on page 123 • Example: Configuring DCBX to Support an iSCSI Application • Understanding DCBX Application Protocol TLV Exchange on page 116 • Understanding DCBX Application Protocol TLV Exchange on EX Series Switches

recommendation-tlv

Syntax	<code>recommendation-tlv { no-auto-negotiation; }</code>
Hierarchy Level	[edit protocols dcbx interface <i>interface-name</i> enhanced-transmission-selection]
Release Information	Statement introduced in Junos OS Release 12.2 for the QFX Series.
Description	Enable DCBX to send the ETS Recommendation TLV (also known as the Information TLV) on egress. This feature is valid only if the interface DCBX mode is IEEE DCBX. If the interface DCBX mode is DCBX version 1.01, this statement has no effect. (DCBX version 1.01 does not advertise separate TLVs for individual attributes.)
Default	DCBX-enabled interfaces send the ETS recommendation TLV unless it is disabled.
Options	no-auto-negotiation —Disable sending of the ETS recommendation TLV.
Required Privilege Level	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• show dcbx neighbors on page 169• Configuring DCBX Autonegotiation on page 112

CHAPTER 7

Operational Commands (FCoE on a Transit Switch)

- clear fip vlan-discovery statistics
- show fip vlan-discovery

clear fip vlan-discovery statistics

Syntax	<code>clear fip vlan-discovery statistics</code>
Release Information	Command introduced in Junos OS Release 12.1 for the QFX Series.
Description	Clear FIP VLAN discovery statistics.
Required Privilege Level	view
Related Documentation	<ul style="list-style-type: none">• show fip vlan-discovery on page 165
List of Sample Output	clear fip vlan-discovery statistics on page 164

Sample Output

clear fip vlan-discovery statistics

```
user@switch> clear fip vlan-discovery statistics
```

show fip vlan-discovery

Syntax	show fip vlan-discovery (enodes statistics)
Release Information	Command introduced in Junos OS Release 12.1 for the QFX Series.
Description	Display FCoE VLAN information from the Fibre Channel switch or FCoE forwarder (FCF).
Options	enodes —Display VLAN discovery information for each ENode. statistics —Display VLAN discovery information statistics.
Required Privilege Level	view
Related Documentation	<ul style="list-style-type: none"> • clear fip vlan-discovery statistics on page 164
List of Sample Output	show fip vlan-discovery enodes on page 166 show fip vlan-discovery statistics (QFX3500) on page 166 show fip vlan-discovery statistics (QFabric Systems) on page 166
Output Fields	Table 13 on page 165 lists the output fields for the show fip vlan-discovery command. Output fields are listed in the approximate order in which they appear.

Table 13: show fip vlan-discovery Output Fields

Field Name	Field Description	Level of Output
Enode-MAC	Media access control (MAC) address of the ENode.	enodes
Interface	Name of the interface.	enodes
Unsolicited notification count	Number of unsolicited VLAN discovery notifications.	All
Solicited notification count	Number of solicited VLAN discovery notifications.	statistics
Node Group Name	Displays the name of the Node group on QFabric systems.	statistics
Request count	Number of VLAN discovery requests sent by the ENode. This number should match the Solicited notification count number.	statistics
VLAN tags	Tags of the FIP-enabled VLANs.	enodes

Sample Output

show fip vlan-discovery enodes

```
user@switch> show fip vlan-discovery enodes
```

Enode-MAC	Interface	Unsolicited Notification Count	Vlan Tags
00:10:94:00:00:02	xe-0/0/9.0	0	400

show fip vlan-discovery statistics (QFX3500)

```
user@switch> show fip vlan-discovery statistics
```

```
Request count: 0  
Solicited notification count: 0  
Unsolicited notification count: 1
```

show fip vlan-discovery statistics (QFabric Systems)

```
user@switch> show fip vlan-discovery statistics
```

```
NW-NG-0:
```

```
-----  
Request count: 0  
Solicited notification count: 0  
Unsolicited notification count: 1
```

```
BBAK0399:
```

```
-----  
Request count: 0  
Solicited notification count: 0  
Unsolicited notification count: 1
```

```
FCC001:
```

```
-----  
Request count: 0  
Solicited notification count: 0  
Unsolicited notification count: 1
```

CHAPTER 8

Operational Commands (DCBX and PFC)

- `show dcbx`
- `show dcbx neighbors`

show dcbx

Syntax	show dcbx
Release Information	Command introduced in Junos OS Release 11.3 for the QFX Series.
Description	List DCBX status (enabled or disabled) and the interfaces on which DCBX is enabled.
Required Privilege Level	view
Related Documentation	<ul style="list-style-type: none"> • show dcbx neighbors on page 169 • Configuring DCBX Autonegotiation on page 112
Output Fields	Table 14 on page 168 lists the output fields for the show dcbx command. Output fields are listed in the approximate order in which they appear.

Table 14: show dcbx output fields

Field Name	Field Description
DCBX	Status of DCBX on the switch or for the specified interface: <ul style="list-style-type: none"> • Enabled—DCBX is enabled on the switch or on the specified interface • Disabled—DCBX is disabled on the switch or on the specified interface
Interface	Name of the interface

Sample Output

show dcbx

```

user@switch> show dcbx
DCBX                : Enabled
Interface           DCBX
xe-0/0/9.0          enabled
xe-0/0/32.0         enabled
xe-0/0/36.0         enabled

```


show dcbx neighbors

Syntax	show dcbx neighbors <interface interface-name> <terse>
Release Information	Command introduced in Junos OS Release 11.1 for the QFX Series. Command introduced in Junos OS Release 11.3 for EX Series switches.
Description	Display information about Data Center Bridging Capability Exchange protocol (DCBX) neighbor interfaces.
Options	none —Display information about all DCBX neighbor interfaces. interface-name —(Optional) Display information for the specified interface. terse —Display the specified level of output.
Required Privilege Level	view
Related Documentation	<ul style="list-style-type: none"> • Configuring DCBX Autonegotiation on page 112 • Example: Configuring DCBX Application Protocol TLV Exchange on page 123 • Example: Configuring an FCoE Transit Switch • Example: Configuring DCBX to Support an iSCSI Application • Understanding DCB Features and Requirements on page 78 • Understanding Data Center Bridging Capability Exchange Protocol for EX Series Switches • dcbx on page 151
List of Sample Output	show dcbx neighbors interface (QFX Series, DCBX Version 1.01 Mode) on page 182 show dcbx neighbors interface (QFX Series, IEEE DCBX Mode) on page 184 show dcbx neighbors terse (QFX Series) on page 186 show dcbx neighbors (EX4500 Switch: FCoE Interfaces on Both Local and Peer with PFC Configured Compatibly) on page 186 show dcbx neighbors (EX4500 Switch: DCBX Interfaces on Local and Peer Are Configured Compatibly with iSCSI Application) on page 187 show dcbx neighbors (EX4500 Switch: Includes ETS) on page 188
Output Fields	Table 15 on page 169 lists the output fields for the show dcbx neighbors command. Output fields are listed in the approximate order in which they appear.

Table 15: show dcbx neighbors Output Fields

Field Name	Field Description
Interface	Name of the interface.

Table 15: show dcbx neighbors Output Fields (*continued*)

Field Name	Field Description
Parent Interface	Name of the link aggregation group (LAG) interface to which the DCBX interface belongs.
Active-application-map	Name of the application map applied to the interface.
Protocol-Mode	<p>(QFX Series) DCBX protocol mode the interface uses:</p> <ul style="list-style-type: none"> IEEE DCBX Version—The interface uses IEEE DCBX mode. DCBX Version 1.01—The interface uses DCBX version 1.01. <p>NOTE: On interfaces that use the IEEE DCBX mode, the show dcbx neighbors interface <i>interface-name</i> operational command does not include application, PFC, or ETS operational state in the output.</p>
Protocol-State	<p>(DCBX Version 1.01 only) DCBX protocol state synchronization status:</p> <ul style="list-style-type: none"> in-sync—The local interface received an acknowledge message from the peer to indicate that the peer received a state change message sent by the local interface. ack-pending—The local interface has not yet received an acknowledge message from the peer to indicate that the peer received a state change message sent by the local interface.
Local-Advertisement	<p>(DCBX Version 1.01 only)</p> <p>Status of advertisements that the local interface sends to the peer.</p>
Operational version	Version of the DCBX standard used.
sequence-number	<p>Number of state change messages sent to the peer.</p> <p>If the interface Protocol-State value is in-sync, this number should match the acknowledge-id number in the Peer-Advertisement section.</p> <p>If the interface Protocol-State value is ack-pending, this number does not match the acknowledge-id number in the Peer-Advertisement section.</p>
acknowledge-id	<p>Number of acknowledge messages received from the peer.</p> <p>If the Protocol-State value is in-sync, this number should match the sequence-number value in the Peer-Advertisement section.</p> <p>If the Protocol-State value is ack-pending, this number does not match the sequence-number value in the Peer-Advertisement section.</p>

Table 15: show dcbx neighbors Output Fields (*continued*)

Field Name	Field Description
Peer-Advertisement	(DCBX Version 1.01 only) Status of advertisements that the peer sends to the local interface.
Operational version	Version of the DCBX standard used.
sequence-number	<p>Number of state change messages the peer sent to the local interface.</p> <p>If this number matches the acknowledge-id number in the Local-Advertisement field, this indicates that the local interface has acknowledged all of the peer's state change messages and is synchronized.</p> <p>If this number does not match the acknowledge-id number in the Local-Advertisement field, this indicates that the peer has not yet received an acknowledgment for a state change message from the local interface.</p>
acknowledge-id	<p>Number of acknowledge messages the peer has received from the local interface.</p> <p>If this number matches the sequence-number value in the Local-Advertisement field, this indicates that the peer has acknowledged all of the local interface's state change messages and is in synchronization.</p> <p>If this number does not match the sequence-number value in the Local-Advertisement field, this indicates that the peer has not yet sent an acknowledgment for a state change message from the local interface.</p>

Table 15: show dcbx neighbors Output Fields (*continued*)

Field Name	Field Description
Feature: PFC	Priority-based flow control (PFC) feature DCBX state information.
Protocol-State	(DCBX Version 1.01 only) DCBX protocol state synchronization status: <ul style="list-style-type: none"> • ack-pending—The local interface has not yet received an acknowledge message from the peer to indicate that the peer received a PFC state change message sent by the local interface. • in-sync—The local interface received an acknowledge message from the peer to indicate that the peer received a PFC state change message sent by the local interface. • not-applicable—PFC autonegotiation is disabled.
Operational State	(DCBX Version 1.01 only) Operational state of the feature: enabled or disabled .
Local-Advertisement	Status of advertisements that the local interface sends to the peer.
Enable	(DCBX Version 1.01 only) State that the local interface advertises to the peer: <ul style="list-style-type: none"> • Yes—The feature is enabled. • No—The feature is disabled.
Willing	Willingness of the local interface to learn the PFC configuration from the peer using DCBX: <ul style="list-style-type: none"> • Yes—The local interface is willing to learn the PFC configuration from the peer. • No—The local interface is not willing to learn the PFC configuration from the peer.
Mac auth Bypass Capability	(IEEE DCBX only) (QFX Series) Media access controller (MAC) authentication bypass provides access to devices based on MAC address authentication. This is not supported, so the only value seen in the local advertisement field is no .
Error	(DCBX Version 1.01 only) Configuration compatibility error status: <ul style="list-style-type: none"> • No—No error detected. Local and peer configuration are compatible. • Yes—Error detected. Local and peer configuration are not compatible.

Table 15: show dcbx neighbors Output Fields (*continued*)

Field Name	Field Description
Operational State	<p>PFC operational state on the interface:</p> <ul style="list-style-type: none"> • Enabled—PFC is enabled on the interface • Disabled—PFC is disabled on the interface
Maximum Traffic Classes capable to support PFC	<p>Largest number of traffic classes the local interface supports for PFC:</p> <ul style="list-style-type: none"> • 6 (EX Series switches) • 6 (QFX Series)
Code Point	<p>PFC code point, which is specified in the 3-bit class-of-service field in the VLAN header.</p>
Admin Mode	<p>PFC administrative state for each code point on the local interface:</p> <ul style="list-style-type: none"> • Enabled—PFC is enabled for the code point. • Disabled—PFC is disabled for the code point.
Operational Mode	<p>(QFX Series) PFC operational mode for each code point:</p> <ul style="list-style-type: none"> • Enable—PFC is enabled on the code point. • Disable—PFC is disabled on the code point.
Peer-Advertisement	<p>Status of advertisements that the peer sends to the local interface.</p>
Enable	<p>(DCBX Version 1.01 only)</p> <p>State that the peer advertises to the local interface:</p> <ul style="list-style-type: none"> • Yes—The feature is enabled. • No—The feature is disabled.
Willing	<p>Willingness of the peer to learn the PFC configuration from the local interface using DCBX:</p> <ul style="list-style-type: none"> • Yes—The peer is willing to learn the PFC configuration from the local interface. • No—The peer is not willing to learn the PFC configuration from the local interface.
Error	<p>(DCBX Version 1.01 only)</p> <p>Configuration compatibility error status:</p> <ul style="list-style-type: none"> • No—No error detected. Local and peer configuration are compatible. • Yes—Error detected. Local and peer configuration are not compatible.

Table 15: show dcbx neighbors Output Fields (*continued*)

Field Name	Field Description
Operational State	<p>PFC operational state on the interface:</p> <ul style="list-style-type: none"> • Enabled—PFC is enabled on the interface • Disabled—PFC is disabled on the interface
Mac auth Bypass Capability	<p>(IEEE DCBX only)</p> <p>(QFX Series) Media access controller (MAC) authentication bypass provides access to devices based on MAC address authentication. Although the QFX Series does not support this feature, the connected peer might support it. This field reports the peer state:</p> <ul style="list-style-type: none"> • Yes—The connected peer supports MAC authentication bypass. • No—The connected peer does not support MAC authentication bypass.
Maximum Traffic Classes capable to support PFC	<p>Largest number of traffic classes the peer supports for PFC:</p> <ul style="list-style-type: none"> • 6 (EX Series switches) • 8 (QFX Series)
Code Point	<p>PFC code point, which is specified in the 3-bit class-of-service field in the VLAN header.</p>
Admin Mode	<p>PFC administrative state for each code point on the peer:</p> <ul style="list-style-type: none"> • Enabled—PFC is enabled for the code point. • Disabled—PFC is disabled for the code point.

Table 15: show dcbx neighbors Output Fields (*continued*)

Field Name	Field Description
Feature: Application	State information for the DCBX application.
Protocol-State	<p>(DCBX Version 1.01 only)</p> <p>DCBX protocol state synchronization status:</p> <ul style="list-style-type: none"> • in-sync—The local interface received an acknowledge message from the peer to indicate that the peer received an FCoE state change message sent by the local interface. • ack-pending—The local interface has not yet received an acknowledge message from the peer to indicate that the peer received an FCoE state change message sent by the local interface. • not-applicable—The local interface is set to no-auto-negotiation (autonegotiation is disabled). If the interface is associated with an FCoE forwarding class, the interface advertises FCoE capability even if the connected peer does not advertise FCoE capability.
Local-Advertisement	<p>Status of advertisements that the local interface sends to the peer.</p> <p>If the local interface is set to no-auto-negotiation (autonegotiation is disabled), the local advertisement portion of the output is not shown.</p>
Enable	<p>(DCBX Version 1.01 only)</p> <p>State that the local interface advertises to the peer:</p> <ul style="list-style-type: none"> • Yes—The feature is enabled. • No—The feature is disabled.
Willing	<p>(DCBX Version 1.01 only)</p> <p>Willingness of the local interface to learn the FCoE interface state from the peer using DCBX:</p> <ul style="list-style-type: none"> • Yes—The local interface is willing to learn the FCoE interface state from the peer. • No—The local interface is not willing to learn the FCoE interface state from the peer.
Error	<p>(DCBX Version 1.01 only)</p> <p>Configuration compatibility error status:</p> <ul style="list-style-type: none"> • No—No error detected. The local and peer configuration are compatible. • Yes—Error detected. The local and peer configuration are not compatible.
Appl-Name	Name of the application:

Table 15: show dcbx neighbors Output Fields (*continued*)

Field Name	Field Description
Ethernet-Type	<p>(DCBX Version 1.01 only)</p> <p>Ethernet type (EtherType) of the application. For example, 0x8906 indicates the EtherType for the FCoE application. Either the EtherType (for Layer 2 applications) or the Socket Number (for Layer 4 applications) of the application is displayed in the output.</p>
Socket-Number	<p>Destination port socket number of the application, if applicable. Either the EtherType (for Layer 2 applications) or the Socket Number (for Layer 4 applications) of the application is displayed in the output.</p>
Priority-Field or Priority-Map	<p>Priority assigned to the application.</p> <p>For EX Series switches, the priority of the FCoE application is determined by the PFC congestion notification profile that has been configured and associated with the FCoE interface. For other applications, the priority is based on the application map.</p>
Status	<p>(DCBX Version 1.01 only)</p> <p>Local status when autonegotiation is enabled:</p> <ul style="list-style-type: none"> • Enabled—The application feature is enabled on both the local interface and the peer interface. (The local configuration and the peer configuration match.) • Disabled—The local configuration and the peer configuration do not match. <p>NOTE: If there is a configuration mismatch in one application between the switch and the peer, all the other applications including FCoE are disabled.</p>
Peer-Advertisement	<p>Status of advertisements that the peer sends to the local interface.</p>
Enable	<p>(DCBX Version 1.01 only)</p> <p>State that the peer advertises to the local interface:</p> <ul style="list-style-type: none"> • Yes—The feature is enabled. • No—The feature is disabled.
Willing	<p>(DCBX Version 1.01 only)</p> <p>Willingness of the peer to learn the FCoE interface state from the local interface using DCBX:</p> <ul style="list-style-type: none"> • Yes—The peer is willing to learn the FCoE interface state from the local interface. • No—The peer is not willing to learn the FCoE interface state from the local interface.

Table 15: show dcbx neighbors Output Fields (*continued*)

Field Name	Field Description
Error	(DCBX Version 1.01 only) Configuration compatibility error status: <ul style="list-style-type: none"> • No—No error detected. Local and peer configuration are compatible. • Yes—Error detected. Local and peer configuration are not compatible.
Appl-Name	Name of the application: <ul style="list-style-type: none"> • FCoE—Fibre Channel over Ethernet
Ethernet-Type	Ethernet type (EtherType) of the application. For example, 0x8906 indicates the EtherType for the FCoE application. Either the EtherType (for Layer 2 applications) or the Socket-Number (for Layer 4 applications) of the application is displayed in the output.
Socket-Number	Destination port socket number of the application, if applicable. Either the EtherType (for Layer 2 applications) or the Socket Number (for Layer 4 applications) of the application is displayed in the output.
Priority-Field or Priority-Map	Priority assigned to the application.
Status	(DCBX Version 1.01 only) Peer interface status: <ul style="list-style-type: none"> • Enabled—The application feature is enabled on both the local interface and the peer interface. (The local configuration and the peer configuration match.) • Disabled—The local configuration and the peer configuration do not match.

Table 15: show dcbx neighbors Output Fields (*continued*)

Field Name	Field Description
Feature: ETS	Enhanced Transmission Selection (ETS) DCBX state information.
Protocol-State	(DCBX Version 1.01 only) ETS protocol state synchronization status: <ul style="list-style-type: none"> • in-sync—The local interface received an acknowledge message from the peer to indicate that the peer received an ETS state change message sent by the local interface. • ack-pending—The local interface has not yet received an acknowledge message from the peer to indicate that the peer received an ETS state change message sent by the local interface.
Operational State	(DCBX Version 1.01 only) Operational state of the feature, enabled or disabled .
Local-Advertisement	Status of advertisements that the local interface sends to the peer.
Enable	(DCBX Version 1.01 only) State that the local interface advertises to the peer: <ul style="list-style-type: none"> • Yes—The feature is enabled. • No—The feature is disabled.
TLV Type	(IEEE DCBX only) Type of ETS TLV: <ul style="list-style-type: none"> • Configuration—Advertises the Configuration TLV, which communicates the local ETS configuration to the peer but does not ask the peer to use the configuration. • Recommendation—Advertises the Recommendation TLV, which communicates the local ETS configuration to the peer, and if the peer is “willing,” configures the peer interface to match the local ETS configuration. • Recommendation-or-Configuration—Advertises both TLVs.
Willing	Willingness of the local interface to learn the ETS state from the peer using DCBX (EX Series switches always advertise No for this field): <ul style="list-style-type: none"> • Yes—Local interface is willing to learn the ETS state from the peer. • No—Local interface is not willing to learn the ETS state from the peer.
Credit Based Shaper	

Table 15: show dcbx neighbors Output Fields (*continued*)

Field Name	Field Description
	(IEEE DCBX only) Alternative method of flow control to buffer-to-buffer credit. The QFX Series does not support a credit-based shaper, so the value of this field is always No .
Error	(DCBX Version 1.01 only) Configuration error status: <ul style="list-style-type: none"> • No—No error. This should always be the switch ETS error state. • Yes—Error detected.
Maximum Traffic Classes capable to support PFC	(DCBX Version 1.01 only) Largest number of traffic classes the local interface supports for PFC.
Maximum Traffic Classes supported	(IEEE DCBX only) Largest number of traffic classes the local interface supports for ETS. (EX Series switches support only one traffic class for ETS. However, a different value might be shown for this field.)
Code Point	PFC code point, which is specified in the 3-bit class-of-service field in the VLAN header.
Priority-Group	Class-of-service (CoS) priority group (forwarding class set) identification number.
Percentage B/W	Configured minimum percentage of link bandwidth allocated to the priority group. Only explicitly configured values appear in this output column. If the link bandwidth is the default percentage, it is not shown. (EX Series switches allocate 100% of link bandwidth to the default priority group, group 7.)
Transmission Selection Algorithm	(IEEE DCBX only) The transmission selection algorithm used by the interface. The QFX Series supports ETS but does not support using the credit-based shaper algorithm, so the only value shown in this field is ETS .
Peer-Advertisement	Status of advertisements that the peer sends to the local interface.
Enable	

Table 15: show dcbx neighbors Output Fields (*continued*)

Field Name	Field Description
	(DCBX Version 1.01 only) State that the peer advertises to the local interface: <ul style="list-style-type: none"> • Yes—The feature is enabled. • No—The feature is disabled.
TLV Type	(IEEE DCBX only) Type of ETS TLV: <ul style="list-style-type: none"> • Configuration—Advertises the Configuration TLV, which communicates the local ETS configuration to the peer but does not ask the peer to use the configuration. • Recommendation—Advertises the Recommendation TLV, which communicates the local ETS configuration to the peer, and if the peer is "willing," configures the peer interface to match the local ETS configuration. • Configuration/Recommendation—Advertises both TLVs.
Willing	Willingness of the peer to learn the ETS state from the local interface using DCBX: <ul style="list-style-type: none"> • Yes—Peer is willing to learn the ETS state from the local interface. • No—Peer is not willing to learn the ETS state from the local interface.
Credit Based Shaper	(IEEE DCBX only) Alternative method of flow control to buffer-to-buffer credit. The QFX Series does not support a credit-based shaper, so the value of this field is always No .
Error	(DCBX Version 1.01 only) Configuration error status of the peer: <ul style="list-style-type: none"> • No—No error in peer ETS TLV. • Yes—Error in peer ETS TLV.
Maximum Traffic Classes capable to support PFC	(DCBX Version 1.01 only) Largest number of traffic classes the local interface supports for PFC.
Maximum Traffic Classes supported	(IEEE DCBX only) Largest number of traffic classes the local interface supports for ETS. (EX Series switches support only one traffic class for ETS. However, a different value might be shown for this field.)
Code Point	

Table 15: show dcbx neighbors Output Fields (*continued*)

Field Name	Field Description
	PFC code point, which is specified in the 3-bit class-of-service field in the VLAN header.
Priority-Group	CoS priority group (forwarding class set) identification number.
Percentage B/W	Configured minimum percentage of link bandwidth allocated to the priority group. (EX Series switches allocate 100% of link bandwidth to the default priority group, group 7.)
Transmission Selection Algorithm	(IEEE DCBX only) Transmission selection algorithm used by the interface. The QFX Series supports ETS but does not support using the credit-based shaper algorithm, so the only value shown in this field is ETS .
PFC	(QFX Series, terse option only) DCBX TLV advertisement state for PFC: <ul style="list-style-type: none"> • Disabled—PFC configuration matches the configuration on the connected peer and PFC is disabled • Enabled—PFC configuration matches the configuration on the connected peer and PFC is enabled • Not Advt—Interface does not advertise PFC to the connected peer
ETS	(terse option only) Local DCBX TLV advertisement state for ETS: <ul style="list-style-type: none"> • Advt—Interface advertises ETS TLVs • Disabled—ETS is disabled on the interface (interface does not advertise ETS)
ETS Rec	(terse option only) DCBX TLV peer advertisement state for ETS (state received from the connected DCBX peer): <ul style="list-style-type: none"> • Advt—Peer interface advertises ETS TLVs • Not Advt—Peer interface does not advertise ETS <p>NOTE: When the DCBX mode is DCBX version 1.01, no peer information is displayed.</p>

Table 15: show dcbx neighbors Output Fields (*continued*)

Field Name	Field Description
Version	<p>(terse option only) The DCBX version used on the interface and whether the DCBX version was autonegotiated or explicitly configured:</p> <ul style="list-style-type: none"> • IEEE—The interface uses IEEE DCBX. • 1.01—The interface uses DCBX version 1.01. <p>When the DCBX version used is the result of autonegotiation, the term (Auto) appears next to the version. For example, IEEE (Auto) indicates that the interface autonegotiated with the connected peer to use IEEE DCBX. Autonegotiation is enabled by default.</p>

Sample Output

show dcbx neighbors interface (QFX Series, DCBX Version 1.01 Mode)

```

user@switch> show dcbx neighbors interface xe-0/0/0
Interface : xe-0/0/0.0 - Parent Interface: ae0.0
Active-application-map: app-map-1
Protocol-State: in-sync
Protocol-Mode: DCBX Version 1.01

Local-Advertisement:
  Operational version: 1
  sequence-number: 130, acknowledge-id: 102

Peer-Advertisement:
  Operational version: 1
  sequence-number: 102, acknowledge-id: 130

Feature: PFC, Protocol-State: in-sync

Operational State: Enabled

Local-Advertisement:
  Enable: Yes, Willing: No, Error: No
  Maximum Traffic Classes capable to support PFC: 8

Code Point      Admin Mode      Operational Mode
000              Disabled        Disable
001              Disabled        Disable
010              Disabled        Disable
011              Enabled         Enable
100              Enabled         Enable
101              Disabled        Disable
110              Disabled        Disable
111              Disabled        Disable

Peer-Advertisement:
  Enable: Yes, Willing: No, Error: No
  Maximum Traffic Classes capable to support PFC: 8

Code Point      Admin Mode
000              Disabled

```

001	Disabled
010	Disabled
011	Enabled
100	Enabled
101	Disabled
110	Disabled
111	Disabled

Feature: Application, Protocol-State: in-sync

Local-Advertisement:

Enable: Yes, Willing: No, Error: No

App1-Name	Ethernet-Type	Socket-Number	Priority-Map	Status
FCoE	0x8906		00001110	Enabled
iSCSI		3260	10000000	Enabled

Peer-Advertisement:

Enable: Yes, Willing: Yes, Error: No

App1-Name	Ethernet-Type	Socket-Number	Priority-Map	Status
FCoE	0x8906	N/A	00001110	Enabled

Feature: ETS, Protocol-State: in-sync

Operational State: Enabled

Local-Advertisement:

Enable: Yes, Willing: No, Error: No

Maximum Traffic Classes capable to support PFC: 8

Code Point	Priority-Group
000	0
001	7
010	7
011	7
100	0
101	1
110	1
111	7

Priority-Group	Percentage B/W
0	40%
1	5%

Peer-Advertisement:

Enable: Yes, Willing: No, Error: No

Maximum Traffic Classes capable to support PFC: 8

Code Point	Priority-Group
000	0
001	7
010	7
011	7
100	0
101	1
110	1

111	7
Priority-Group	Percentage B/W
0	40%
1	5%

show dcbx neighbors interface (QFX Series, IEEE DCBX Mode)

```
user@switch> show dcbx neighbors interface xe-0/0/0
```

```
Interface : xe-0/0/0.0 - Parent Interface: ae0.0
```

```
Active-application-map: app-map-1
```

```
Protocol-Mode: IEEE-DCBX Version
```

Feature: PFC

Local-Advertisement:

Willing: No

Mac auth Bypass Capability: No

Operational State: Enabled

Maximum Traffic Classes capable to support PFC: 8

Code Point	Admin Mode
000	Disabled
001	Disabled
010	Disabled
011	Enabled
100	Enabled
101	Disabled
110	Disabled
111	Disabled

Peer-Advertisement:

Willing: No

Mac auth Bypass Capability: No

Operational State: Enabled

Maximum Traffic Classes capable to support PFC: 8

Code Point	Admin Mode
000	Disabled
001	Disabled
010	Disabled
011	Enabled
100	Enabled
101	Disabled
110	Disabled
111	Disabled

Feature: Application

Local-Advertisement:

App1-Name	Ethernet-Type	Socket-Number	Priority-field
FCoE	0x8906		00001110
iSCSI		3260	10000000

Peer-Advertisement:

App1-Name	Ethernet-Type	Socket-Number	Priority-field
-----------	---------------	---------------	----------------

FCoE	0x8906	N/A	00001110
------	--------	-----	----------

Feature: ETS

Local-Advertisement:

TLV Type: Configuration/Recommendation

Willing: No

Credit Based Shaper: No

Maximum Traffic Classes supported: 3

Code Point	Priority-Group
000	0
001	7
010	7
011	7
100	0
101	1
110	1
111	7

Priority-Group	Percentage B/W
0	40%
1	5%

Priority-Group	Transmission Selection Algorithm
0	Enhanced Transmission Selection
1	Enhanced Transmission Selection

Peer-Advertisement:

TLV Type: Configuration

Willing: No

Credit Based Shaper: No

Code Point	Priority-Group
000	0
001	7
010	7
011	7
100	0
101	1
110	1
111	7

Priority-Group	Percentage B/W
0	40%
1	5%

Priority-Group	Transmission Selection Algorithm
0	Enhanced Transmission Selection
1	Enhanced Transmission Selection

Peer-Advertisement:

TLV Type: Recommendation

Code Point	Priority-Group
000	0
001	7
010	7
011	7
100	0

101	1
110	1
111	7
Priority-Group	Percentage B/W
0	40%
1	5%
Priority-Group	Transmission Selection Algorithm
0	Enhanced Transmission Selection
1	Enhanced Transmission Selection

show dcbx neighbors terse (QFX Series)

```

user@switch> show dcbx neighbors terse
Interface Parent PFC ETS ETS Version
Interface Rec
xe-0/0/8.0 - Enabled Advt Advt IEEE (Auto)
xe-0/0/9.0 - Disabled Disabled 1.01
xe-0/0/11.0 ae0.0 Enabled Advt Advt IEEE (Auto)
xe-0/0/12.0 ae0.0 Enabled Advt Advt IEEE (Auto)
xe-0/0/32.0 - Enabled Advt Not Advt IEEE
xe-0/0/36.0 - Not Advt Advt Advt IEEE

```

show dcbx neighbors (EX4500 Switch: FCoE Interfaces on Both Local and Peer with PFC Configured Compatibly)

```

user@switch> show dcbx neighbors interface xe-0/0/14

Interface : xe-0/0/14.0 - Parent Interface: ae0.0
Protocol-State: in-sync

Local-Advertisement:
  Operational version: 0
  sequence-number: 6, acknowledge-id: 6

Peer-Advertisement:
  Operational version: 0
  sequence-number: 6, acknowledge-id: 6

Feature: PFC, Protocol-State: in-sync

Operational State: Enabled

Local-Advertisement:
  Enable: Yes, Willing: No, Error: No
  Maximum Traffic Classes capable to support PFC: 6

Code Point      Admin Mode
000             Disabled
001             Disabled
010             Disabled
011             Enabled
100             Disabled
101             Disabled
110             Disabled
111             Disabled

```

Peer-Advertisement:

Enable: Yes, Willing: No, Error: No

Maximum Traffic Classes capable to support PFC: 6

Code Point	Admin Mode
000	Disabled
001	Disabled
010	Disabled
011	Enabled
100	Disabled
101	Disabled
110	Disabled
111	Disabled

Feature: Application, Protocol-State: in-sync

Local-Advertisement:

Enable: Yes, Willing: No, Error: No <<< Error bit will not be set as there is no miss configuration between local and peer.

Appl-Name	Ethernet-Type	Socket-Number	Priority-Map	Status
FCoE	0x8906		00001000	Enabled

Peer-Advertisement:

Enable: Yes, Willing: No, Error: No

Status	Appl-Name	Ethernet-Type	Socket-Number	Priority-Map
Enabled	FCoE	0x8906		00001000

show dcbx neighbors (EX4500 Switch: DCBX Interfaces on Local and Peer Are Configured Compatibly with iSCSI Application)

user@switch> show dcbx neighbors interface xe-0/0/14

Interface : xe-0/0/14.0 - Parent Interface: ae0.0

Protocol-State: in-sync

Active-application-map: iscsi-map

Local-Advertisement:

Operational version: 0

sequence-number: 9, acknowledge-id: 12

Peer-Advertisement:

Operational version: 0

sequence-number: 12, acknowledge-id: 9

Feature: PFC, Protocol-State: in-sync

Operational State: Enabled

Local-Advertisement:

Enable: Yes, Willing: No, Error: No

Maximum Traffic Classes capable to support PFC: 6

Code Point	Admin Mode
000	Disabled
001	Disabled
010	Disabled
011	Enabled
100	Disabled
101	Disabled
110	Disabled
111	Disabled

Peer-Advertisement:

Enable: Yes, Willing: No, Error: No

Maximum Traffic Classes capable to support PFC: 6

Code Point	Admin Mode
000	Disabled
001	Disabled
010	Disabled
011	Enabled
100	Disabled
101	Disabled
110	Disabled
111	Disabled

Feature: Application, Protocol-State: in-sync

Local-Advertisement:

Enable: Yes, Willing: No, Error: No

Appl-Name	Ethernet-Type	Socket-Number	Priority-Map	Status
FCoE	0x8906		00001000	Enabled
iscsi		3260	00100000	Enabled

Peer-Advertisement:

Enable: Yes, Willing: No, Error: No

Appl-Name	Ethernet-Type	Socket-Number	Priority-Map	Status
FCoE	0x8906		00001000	Enabled
iscsi		3260	00100000	Enabled

show dcbx neighbors (EX4500 Switch: Includes ETS)

user@switch> show dcbx neighbors interface xe-0/0/3

Interface : xe-0/0/3.0
 Protocol-State: in-sync
 Active-application-map: map_iscsi

Local-Advertisement:

Operational version: 0

sequence-number: 1, acknowledge-id: 5

Peer-Advertisement:

Operational version: 0

sequence-number: 5, acknowledge-id: 1

Feature: PFC, Protocol-State: in-sync

Operational State: Enabled

Local-Advertisement:

Enable: Yes, Willing: No, Error: No

Maximum Traffic Classes capable to support PFC: 6

Code Point	Admin Mode
000	Enabled
001	Enabled
010	Disabled
011	Disabled
100	Disabled
101	Disabled
110	Disabled
111	Disabled

Peer-Advertisement:

Enable: Yes, Willing: Yes, Error: No

Maximum Traffic Classes capable to support PFC: 8

Code Point	Admin Mode
000	Enabled
001	Disabled
010	Disabled
011	Disabled
100	Enabled
101	Disabled
110	Disabled
111	Disabled

Feature: Application, Protocol-State: in-sync

Local-Advertisement:

Enable: Yes, Willing: No, Error: No

App1-Name	Ethernet-Type	Socket-Number	Priority-Map	Status
FCoE	0x8906		00000001	Enabled
iscsi		3260	00000010	Enabled

Peer-Advertisement:

Enable: Yes, Willing: Yes, Error: No

App1-Name	Ethernet-Type	Socket-Number	Priority-Map	Status
FCoE	0x8906		00010000	Enabled
iscsi		3260	00010000	Enabled

Feature: ETS, Protocol-State: in-sync

Operational State: Enabled

Local-Advertisement:

Enable: Yes, Willing: No, Error: No
Maximum Traffic Classes supported : 3

Code Point	Priority-Group
000	7
001	7
010	7
011	7
100	7
101	7
110	7
111	7

Priority-Group	Percentage B/W
7	100%

Peer-Advertisement:

Enable: Yes, Willing: Yes, Error: No
Maximum Traffic Classes supported : 8

Code Point	Priority-Group
000	0
001	1
010	0
011	0
100	2
101	0
110	0
111	0

Priority-Group	Percentage B/W
0	30%
1	40%
2	30%