Junos® OS for EX Series Ethernet Switches

Storage on EX4600

Release
13.2X51

Published: 2014-07-30

Copyright © 2014, Juniper Networks, Inc.
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 ......................................................... xiii
  Documentation and Release Notes ........................................... xiii
  Supported Platforms ......................................................... xiii
  Using the Examples in This Manual ........................................ xiii
    Merging a Full Example .................................................. xiv
    Merging a Snippet ....................................................... xiv
  Documentation Conventions ............................................... xv
  Documentation Feedback ..................................................... xvii
  Requesting Technical Support ............................................. xvii
  Self-Help Online Tools and Resources .................................. xvii
  Opening a Case with JTAC .................................................. xviii

Part 1  Overview

Chapter 1  Software Features Overview .................................... 3
  Overview of Fibre Channel .................................................. 4
  Fibre Channel Transport Protocol ......................................... 5
  How FC Works on the Switch ................................................ 5
    FCoE-FC Gateway ........................................................... 5
    FCoE Transit Switch ...................................................... 6
    FCoE VLANs ............................................................... 6
  Supported FC Features and Functions .................................... 8
  Lossless Transport Support .................................................. 8
  Overview of FIP ............................................................... 9

Chapter 2  Fibre Channel, FCoE, FIP, and FIP Snooping ...................... 11
  Understanding Fibre Channel .............................................. 12
  FC Fabrics ........................................................................ 12
  FC Port Types ..................................................................... 13
  FC Switches ....................................................................... 13
  Adapters ............................................................................ 13
  N_Port ID Virtualization (NPIV) ............................................. 14
  FC Services ....................................................................... 14
  Understanding DCB Features and Requirements ......................... 16
  Lossless Transport ............................................................. 16
    PFC ............................................................................. 17
    Buffer Management .......................................................... 17
    Physical Interfaces .......................................................... 17
  ETS .................................................................................. 17
  DCBX .............................................................................. 18
Part 2 Configuration

Chapter 4 Configuration Examples ................................................. 105
Example: Configuring DCBX Application Protocol TLV Exchange .......... 105
Example: Configuring CoS PFC for FCoE Traffic .......................... 115

Chapter 5 Configuration Examples (ELS CLI for Platforms that Support FCoE Only) ........................................... 125
Example: Configuring CoS for FCoE Transit Switch Traffic Across an MC-LAG 125
Example: Configuring VN2VN_Port FIP Snooping (FCoE Hosts Directly Connected to the Same FCoE Transit Switch) ... 147
Example: Configuring VN2VN_Port FIP Snooping (FCoE Hosts Directly Connected to Different FCoE Transit Switches) ... 152
Example: Configuring VN2VN_Port FIP Snooping (FCoE Hosts Indirectly Connected Through an Aggregation Layer FCoE Transit Switch) ... 159

Chapter 6 Configuration Tasks (Fibre Channel, FCoE, FIP, and FIP Snooping) .... 169
Configuring a Physical Fibre Channel Interface ................................ 170
Enabling and Disabling CoS OxID Hash Control on Standalone Switches ... 171
Setting the Maximum Number of FIP Login Sessions per ENode ............ 172
Setting the Maximum Number of FIP Login Sessions per Node Device .... 173
Configuring VLANs for FCoE Traffic on an FCoE Transit Switch .......... 174
Configuring VN2VF_Port FIP Snooping and FCoE Trusted Interfaces on an FCoE Transit Switch ............................................ 179
Enabling VN2VN_Port FIP Snooping and Configuring the Beacon Period on an FCoE Transit Switch ...................... 182

Chapter 7 Configuration Tasks (DCBX) ............................................. 185
Configuring the DCBX Mode ..................................................... 185
Configuring DCBX Autonegotiation ............................................. 186
Disabling the ETS Recommendation TLV .................................. 189
Defining an Application for DCBX Application Protocol TLV Exchange ... 189
Configuring an Application Map for DCBX Application Protocol TLV Exchange ................................................................. 191
Applying an Application Map to an Interface for DCBX Application Protocol TLV Exchange ....................................................... 192

Chapter 8 Configuration Statements ............................................. 193
application (Application Maps) .................................................. 194
application (Applications) ....................................................... 195
application-map ............................................................... 196
application-maps ............................................................ 197
applications (Applications) ................................................... 198
Chapter 9 Configuration Statements (ELS CLI for Platforms that Support FCoE Only) .......................... 219
examine-vn2vf .......................................................... 220
fip-security .............................................................. 221
interface (FIP Snooping) ............................................. 222

Part 3 Administration
Chapter 10 Operational Commands .................................. 225
clear fibre-channel fc2 statistics ..................................... 227
clear fibre-channel fip enode ......................................... 228
clear fibre-channel fip statistics ..................................... 229
clear fibre-channel fip vn-port ......................................... 230
clear fibre-channel flogi statistics ................................... 231
clear fibre-channel proxy statistics ................................... 232
clear fip snooping enode ............................................... 233
clear fip snooping statistics ........................................... 234
clear fip snooping vlan ................................................ 235
clear fip vlan-discovery statistics .................................... 236
restart ................................................................. 237
show dcbx .............................................................. 248
show dcbx neighbors .................................................. 249
show fibre-channel fabric ............................................. 271
show fibre-channel fc2 sessions ..................................... 273
show fibre-channel fc2 sessions ..................................... 275
show fibre-channel fip .................................................. 277
show fibre-channel fip enode ......................................... 282
show fibre-channel fip fabric ......................................... 286
show fibre-channel fip fc ............................................... 289
show fibre-channel fip interface ..................................... 292
show fibre-channel fip statistics ..................................... 295
show fibre-channel flogi fport . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 299
show fibre-channel flogi nport . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 301
show fibre-channel flogi statistics . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 303
show fibre-channel interfaces . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 306
show fibre-channel next-hops . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 309
show fibre-channel routes . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 311
show fibre-channel proxy fabric-state . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 313
show fibre-channel proxy login-table . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 317
show fibre-channel proxy np-port . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 320
show fibre-channel proxy statistics . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 323
show fip snooping . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 326
show fip snooping enode . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 331
show fip snooping fcf . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 335
show fip snooping interface . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 338
show fip snooping statistics . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 341
show fip snooping vlan . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 344
show fip vlan-discovery . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 348
show route forwarding-table family fibre-channel . . . . . . . . . . . . . . . . . . . . . . . . . . 350

Part 4
Troubleshooting

Chapter 11
Troubleshooting Procedures . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 355
Troubleshooting Dropped FCoE Traffic . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 355
Troubleshooting Dropped FIP Traffic . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 358
# List of Figures

## Part 1  Overview

**Chapter 2  Fibre Channel, FCoE, FIP, and FIP Snooping**

- Figure 1: ENode Components ........................................ 22
- Figure 2: FCoE Transit Switch Connecting FCoE Devices to an FC Switch .......... 28
- Figure 3: FCoE Transit Switch Performs VN2VF Port FIP Snooping .............. 39
- Figure 4: VN2VN Port Traffic Across a QFabric Interconnect Device .......... 51
- Figure 5: Supported Topology for an MC-LAG on an FCoE Transit Switch ...... 63

## Part 2  Configuration

**Chapter 4  Configuration Examples** .................................................. 105

- Figure 6: PFC for FCoE Traffic Configuration Components Block Diagram ..... 118

**Chapter 5  Configuration Examples (ELS CLI for Platforms that Support FCoE Only)** ................................................................. 125

- Figure 7: Supported Topology for an MC-LAG on an FCoE Transit Switch .... 127
- Figure 8: VN2VN Port FIP Snooping (FCoE Hosts Connected to Same Transit Switch) Topology ........................................ 150
- Figure 9: VN2VN Port FIP Snooping (FCoE Hosts Connected to Different Transit Switches) Topology ........................................ 155
- Figure 10: VN2VN Port FIP Snooping (FCoE Hosts Indirectly Connected) Topology ........................................ 162
List of Tables

About the Documentation ....................................................... xiii
Table 1: Notice Icons ............................................................. xv
Table 2: Text and Syntax Conventions ....................................... xv

Part 1 Overview
Chapter 1 Software Features Overview .................................. 3
Table 3: Fibre Channel Protocol Layers ..................................... 5
Chapter 2 Fibre Channel, FCoE, FIP, and FIP Snooping .......... 11
Table 4: VFP TCAM Entry Consumption Summary .................... 57
Table 5: Asymmetric Ethernet PAUSE Flow Control Configuration . 68
Table 6: Flow Control State Advertised to the Connected Peer
(Autonegotiation) ................................................................. 69
Table 7: Asymmetric Ethernet PAUSE Behavior on Local and Peer Interfaces . 70
Table 8: Default PFC Priority to Queue and Forwarding Class Mapping . 72
Table 9: Fibre Channel Terms .................................................. 76

Chapter 3 DCBX ................................................................. 87
Table 10: Summary of Differences Between IEEE DCBX and DCBX Version 1.01 ................................................................. 89

Part 2 Configuration
Chapter 4 Configuration Examples ......................................... 105
Table 11: Default IEEE 802.1 Classifiers for Trunk Ports and Tagged-Access Ports
(Default Trusted Classifier) ................................................... 107
Table 12: Default IEEE 802.1 Unicast Classifiers for Access Ports (Default Untrusted Classifier) ................................................... 107
Table 13: Components of DCBX Application Protocol Exchange Configuration Topology .................................................. 108
Table 14: Components of the PFC for FCoE Traffic Configuration Topology ................................................................. 117

Chapter 5 Configuration Examples (ELS CLI for Platforms that Support FCoE Only) ................................................................. 125
Table 15: Components of the CoS for FCoE Traffic Across an MC-LAG Configuration Topology .................................................. 127
Table 16: Components of the VN2VN_Port FIP Snooping Configuration Topology (FCoE Hosts Directly Connected to the Same FCoE Transit Switch) ................................................... 149
Table 17: Components of the VN2VN_Port FIP Snooping Configuration Topology (FCoE Hosts Directly Connected to Different FCoE Transit Switches) ................................................... 154
Table 18: Components of the VN2VN_Port FIP Snooping Configuration Topology
(FCoE Hosts Indirectly Connected Across an Aggregation Layer FCoE Transit Switch) ......................................................... 161

Part 3
Chapter 10

Operational Commands .............................................................. 225

Table 19: show dcbx output fields ................................................. 248
Table 20: show dcbx neighbors Output Fields .............................. 249
Table 21: show fibre-channel fabric Output Fields ......................... 271
Table 22: show fibre-channel fc2 sessions Output Fields ................. 273
Table 23: show fibre-channel fc2 statistics Output Fields ............... 275
Table 24: show fibre-channel fip Output Fields ............................ 277
Table 25: show fibre-channel fip enode Output Fields ................... 282
Table 26: show fibre-channel fip fabric Output Fields ................. 286
Table 27: show fibre-channel fip fcf Output Fields ....................... 289
Table 28: show fibre-channel fip interface Output Fields ............... 292
Table 29: show fibre-channel fip statistics Output Fields ............... 295
Table 30: show fibre-channel flogi fport Output Fields ................. 299
Table 31: show fibre-channel flogi nport Output Fields ................. 301
Table 32: show fibre-channel flogi statistics Output Fields .......... 303
Table 33: show fibre-channel interfaces Output Fields ................. 306
Table 34: show fibre-channel next-hops Output Fields ................. 309
Table 35: show fibre-channel routes Output Fields ...................... 311
Table 36: show fibre-channel proxy fabric-state Output Fields ....... 313
Table 37: show fibre-channel proxy login-table Output Fields ......... 317
Table 38: show fibre-channel proxy np-port Output Fields .......... 320
Table 39: show fibre-channel proxy statistics Output Fields ......... 323
Table 40: show fip snooping Output Fields ................................. 326
Table 41: show fip snooping enode Output Fields ....................... 331
Table 42: show fip snooping fcf Output Fields ............................ 335
Table 43: show fip snooping interface Output Fields ................. 338
Table 44: show fip snooping statistics Output Fields .................. 341
Table 45: show fip snooping vlan Output Fields ......................... 344
Table 46: show fip vlan-discovery Output Fields ......................... 348
Table 47: show route forwarding-table family fibre-channel Output Fields .............................. 351
About the Documentation

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

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:

- EX 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.xsl;
       }
     }
   }

   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.xsl;
   } 
   ```

2. Move to the hierarchy level that is relevant for this snippet by issuing the following configuration mode command:
3. Merge the contents of the file into your routing platform configuration by issuing the `load merge relative` configuration mode command:

```bash
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 xv defines notice icons used in this guide.

<table>
<thead>
<tr>
<th>Icon</th>
<th>Meaning</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="https://example.com/info-icon.png" alt="Info" /></td>
<td>Informational note</td>
<td>Indicates important features or instructions.</td>
</tr>
<tr>
<td><img src="https://example.com/caution-icon.png" alt="Caution" /></td>
<td>Caution</td>
<td>Indicates a situation that might result in loss of data or hardware damage.</td>
</tr>
<tr>
<td><img src="https://example.com/warning-icon.png" alt="Warning" /></td>
<td>Warning</td>
<td>Alerts you to the risk of personal injury or death.</td>
</tr>
<tr>
<td><img src="https://example.com/laser-icon.png" alt="Laser" /></td>
<td>Laser warning</td>
<td>Alerts you to the risk of personal injury from a laser.</td>
</tr>
<tr>
<td><img src="https://example.com/tip-icon.png" alt="Tip" /></td>
<td>Tip</td>
<td>Indicates helpful information.</td>
</tr>
<tr>
<td><img src="https://example.com/best-practice-icon.png" alt="Best practice" /></td>
<td>Best practice</td>
<td>Alerts you to a recommended use or implementation.</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Convention</th>
<th>Description</th>
<th>Examples</th>
</tr>
</thead>
</table>
| **Bold text like this** | Represents text that you type. | To enter configuration mode, type the `configure` command:  
  ```bash
  user@host> configure
  ``` |
### Table 2: Text and Syntax Conventions (continued)

<table>
<thead>
<tr>
<th>Convention</th>
<th>Description</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fixed-width text like this</td>
<td>Represents output that appears on the terminal screen.</td>
<td><code>user@host&gt; show chassis alarms</code>&lt;br&gt;No alarms currently active</td>
</tr>
<tr>
<td><strong>Italic text like this</strong></td>
<td></td>
<td>• A policy term is a named structure that defines match conditions and actions.&lt;br&gt;• <em>Junos OS CLI User Guide</em>&lt;br&gt;• <em>RFC 1997, BGP Communities Attribute</em></td>
</tr>
<tr>
<td><strong>Italic text like this</strong></td>
<td></td>
<td>• Represents variables (options for which you substitute a value) in commands or configuration statements.</td>
</tr>
<tr>
<td>Text like this</td>
<td>Represents names of configuration statements, commands, files, and directories; configuration hierarchy levels; or labels on routing platform components.</td>
<td>• To configure a stub area, include the <code>stub</code> statement at the <code>[edit protocols ospf area area-id]</code> hierarchy level.&lt;br&gt;• The console port is labeled <code>CONSOLE</code>.</td>
</tr>
<tr>
<td><code>&lt; &gt;</code> (angle brackets)</td>
<td>Encloses optional keywords or variables.</td>
<td><code>stub &lt;default-metric metric&gt;;</code></td>
</tr>
<tr>
<td>`</td>
<td>` (pipe symbol)</td>
<td>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.</td>
</tr>
<tr>
<td><code>#</code> (pound sign)</td>
<td>Indicates a comment specified on the same line as the configuration statement to which it applies.</td>
<td><code>rsvp [&lt;default-metric metric&gt;];</code></td>
</tr>
<tr>
<td><code>[ ]</code> (square brackets)</td>
<td>Encloses a variable for which you can substitute one or more values.</td>
<td><code>community name members [community-ids ]</code></td>
</tr>
</tbody>
</table>
| Indentation and braces ( `{ }` ) | Identifies a level in the configuration hierarchy. | `[edit]<br>routing-options {<br>static {
  route default {
    nexthop address;
    retain;
  }
}
}                                                 |
| `:` (semicolon)         | Identifies a leaf statement at a configuration hierarchy level.              |                                                                                               |
| **GUI Conventions**     |                                                                             | • In the Logical Interfaces box, select All Interfaces.<br>• To cancel the configuration, click Cancel. |
Table 2: Text and Syntax Conventions (continued)

<table>
<thead>
<tr>
<th>Convention</th>
<th>Description</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt; (bold right angle bracket)</td>
<td>Separates levels in a hierarchy of menu selections.</td>
<td>In the configuration editor hierarchy, select Protocols &gt; Ospf.</td>
</tr>
</tbody>
</table>

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 at the Juniper Networks Technical Documentation site at [http://www.juniper.net/techpubs/index.html](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 [https://www.juniper.net/cgi-bin/docbugreport/](https://www.juniper.net/cgi-bin/docbugreport/).

- 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 JNASC 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.


- Product warranties—For product warranty information, visit [http://www.juniper.net/support/warranty/](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/](http://www.juniper.net/customers/support/)

- Search for known bugs: [http://www2.juniper.net/kb/](http://www2.juniper.net/kb/)

- Find product documentation: [http://www.juniper.net/techpubs/](http://www.juniper.net/techpubs/)

- Find solutions and answer questions using our Knowledge Base: [http://kb.juniper.net/](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.
PART 1

Overview

- Software Features Overview on page 3
- Fibre Channel, FCoE, FIP, and FIP Snooping on page 11
- DCBX on page 87
CHAPTER 1

Software Features Overview

- Overview of Fibre Channel on page 4
- Overview of FIP on page 9
Overview of Fibre Channel

Fibre Channel (FC) is a high-speed network technology that interconnects network elements and allows them to communicate with one another. The International Committee for Information Technology Standards (INCITS) T11 Technical Committee sets FC standards.

FC networks provide high-performance characteristics such as lossless transport combined with flexible network topology. FC is primarily used in storage area networks (SANs) because it provides reliable, lossless, in-order frame transport between initiators and targets. FC components include initiators, targets, and FC-capable switches that interconnect FC devices and may also interconnect FC devices with Fibre Channel over Ethernet (FCoE) devices. Initiators originate I/O commands. Targets receive I/O commands. For example, a server can initiate an I/O request to a storage device target.

The Juniper Networks QFX3500 Switch has native FC ports as well as Ethernet access ports, and can function as an FCoE-FC gateway or as an FCoE transit switch. All other QFX Series switches and EX4600 switches have Ethernet access ports and can function as an FCoE transit switch.

FCoE transports native FC frames over an Ethernet network by encapsulating the unmodified frames in Ethernet. It also provides protocol extensions to discover FCoE devices through the Ethernet network. FCoE requires that the Ethernet network support data center bridging (DCB) extensions that ensure lossless transport and allow the Layer 2 Ethernet domain to meet the requirements of FC transport.

The FCoE-FC gateway functionality is a licensed feature on the QFX Series that is available only on QFX3500 switches. As an FCoE-FC gateway, the switch connects FCoE devices on an Ethernet network to a SAN FC switch.

You do not need a license to use the switch as an FCoE transit switch. As an FCoE transit switch, the switch:

- Is a Layer 2 data center bridging (DCB) switch that can transport FCoE frames.
- Implements FCoE Initialization Protocol (FIP) snooping.
- Connects multiple FCoE endpoints to the FC network.

NOTE: Only pure QFX5100 Virtual Chassis (VC) configurations (configurations that consist only of QFX5100 switches) and pure QFX5100 Virtual Chassis Fabric (VCF) configurations support FCoE. Mixed mode and other single-platform VCs and VCFs do not support FCoE.

This topic describes:

- Fibre Channel Transport Protocol on page 5
- How FC Works on the Switch on page 5
Fibre Channel Transport Protocol

The Fibre Channel Protocol is a transport protocol that consists of five layers as shown in Table 3 on page 5:

<table>
<thead>
<tr>
<th>FC Protocol Layer</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FC-0</td>
<td>Physical (cabling, connectors, and so on)</td>
</tr>
<tr>
<td>FC-1</td>
<td>Data link layer</td>
</tr>
<tr>
<td>FC-2</td>
<td>Network layer (defines the main protocols)</td>
</tr>
<tr>
<td>FC-3</td>
<td>Common services</td>
</tr>
<tr>
<td>FC-4</td>
<td>Protocol mapping</td>
</tr>
</tbody>
</table>

The FC protocol layers are generally split into three groups:

• FC-0 and FC-1 are the physical layers.
• FC-2 is the protocol layer, similar to OSI Layer 3.
• FC-3 and FC-4 are the services layers.

The FCoE-FC gateway operates the physical layers and the protocol layer, and provides FIP and service redirection at the services layer.

How FC Works on the Switch

The switch connects devices that support FC and Ethernet (such as FCoE servers on an Ethernet network) to an FC SAN, thus converging the Ethernet and FC networks on a single physical network infrastructure. The switch provides the class-of-service (CoS) features needed to handle the different types of traffic appropriately.

To converge FC and Ethernet networks, you can configure the switch as an:

• FCoE-FC Gateway on page 5
• FCoE Transit Switch on page 6
• FCoE VLANs on page 6

FCoE-FC Gateway

When the switch functions as an FCoE-FC gateway, the switch aggregates FCoE traffic and performs the encapsulation and de-encapsulation of native FC frames in Ethernet as it transports the frames between FCoE devices in the Ethernet network and the FC switch. In effect, the switch translates Ethernet to FC and FC to Ethernet.
The gateway receives FC frames encapsulated in Ethernet from FCoE devices through an FCoE VLAN interface composed of one or more 10-Gigabit Ethernet interfaces. The gateway removes the Ethernet encapsulation from the FC frames, and then sends the native FC frames to the FC switch through a native FC interface.

The gateway receives native FC frames from the FC switch on the gateway’s native FC interfaces. The gateway encapsulates the native FC frames in Ethernet, and then sends the encapsulated frames to the appropriate FCoE device through the FCoE VLAN interface.

To FCoE devices, the gateway behaves like an FC switch and can present multiple virtual F_Ports (VF_Ports) on a single interface. To an FC switch, the gateway behaves like an FC node that is doing N_Port ID virtualization (NPIV).

### FCoE Transit Switch

When the switch functions as an FCoE transit switch, it forwards traffic (including FCoE traffic) based on Layer 2 media access control (MAC) forwarding and is a normal DCB-enabled Layer 2 switch that also performs FIP snooping. The switch aggregates FCoE traffic and passes it through to an FCF. The switch does not remove the Ethernet encapsulation from the FC frames, but it does preserve the class of service (CoS) required to transport FC frames.

The switch inspects (snoops) FIP information in order to create filters that permit only valid FCoE traffic to flow through the switch between FCoE devices and the FCF. The switch does not use native FC ports because the FC frames are encapsulated in Ethernet when they flow between the FCoE devices and the FCF. Virtual point-to-point links between each FCoE device and the FCF pass transparently through the switch, so the switch is not seen as a terminating point or an intermediate point by FCoE devices or by the FCF.

### 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:** 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.

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.
**Supported FC Features and Functions**

The following features and functionality are supported:

- **As an FCoE-FC gateway:**
  - DCB, including Data Center Bridging Capability Exchange protocol (DCBX), priority-based flow control (PFC), enhanced transmission service (ETS), and 10-Gigabit Ethernet interfaces
  - FCoE Initialization Protocol (FIP)
  - Proxy for FCoE devices when communicating with FC switches and acts as a proxy for FC switches when communicating with FCoE devices
  - Up to 12 native FC interfaces per QFX3500 switch (each interface can be configured as a 2-Gigabit, 4-Gigabit, or 8-Gigabit Ethernet interface)

- **As an FCoE transit switch:**
  - DCB functions
  - FIP snooping
  - Transparent Layer 2 MAC forwarding of FCoE frames

**Lossless Transport Support**

Up to six lossless forwarding classes are supported. For lossless transport, you must enable PFC on the IEEE 802.1p code point of lossless forwarding classes. The following limitations apply to support lossless transport:

- The external cable length from a standalone switch or QFabric system Node device to other devices cannot exceed 300 meters.
- The internal cable length from a QFabric system Node device to the QFabric system Interconnect device cannot exceed 150 meters.
- For FCoE traffic, the interface maximum transmission unit (MTU) must be at least 2180 bytes to accommodate the packet payload, headers, and checks.

**Related Documentation**

- Understanding Fibre Channel on page 12
- Understanding an FCoE-FC Gateway
- Understanding FCoE Transit Switch Functionality on page 26
- Understanding FCoE on page 20
- Understanding DCB Features and Requirements on page 16
- Overview of FIP on page 9
- Understanding VN_Port to VF_Port FIP Snooping on an FCoE Transit Switch on page 38
- Understanding CoS Flow Control (Ethernet PAUSE and PFC) on page 66
- Understanding Interfaces on an FCoE-FC Gateway
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.


Related Documentation

- Overview of Fibre Channel on page 4
- Understanding Fibre Channel on page 12
- Understanding FIP Functions on page 32
- Understanding FIP Implementation on an FCoE-FC Gateway
- Understanding FIP Parameters on an FCoE-FC Gateway
- Understanding Fibre Channel Virtual Links on page 36
- Understanding FCoE on page 20
- Understanding an FCoE-FC Gateway
- Configuring FIP on an FCoE-FC Gateway
- Understanding Fibre Channel Terminology on page 76
CHAPTER 2

Fibre Channel, FCoE, FIP, and FIP Snooping

• Understanding Fibre Channel on page 12
• Understanding DCB Features and Requirements on page 16
• Understanding FCoE on page 20
• Understanding FCoE Transit Switch Functionality on page 26
• Understanding FCoE and FIP Session High Availability on page 30
• Understanding FIP Functions on page 32
• Understanding Fibre Channel Virtual Links on page 36
• Understanding OXID Hash Control for FCoE Traffic Load Balancing on Standalone Switches on page 37
• Understanding VN_Port to VF_Port FIP Snooping on an FCoE Transit Switch on page 38
• Understanding VN_Port to VN_Port FIP Snooping on an FCoE Transit Switch on page 45
• Understanding FIP Snooping, FBF, and MVR Filter Scalability on page 53
• Understanding MC-LAGs on an FCoE Transit Switch on page 62
• Understanding CoS Flow Control (Ethernet PAUSE and PFC) on page 66
• Understanding Fibre Channel Terminology on page 76
Understanding Fibre Channel

Fibre Channel (FC) is a serial I/O interconnect network technology capable of supporting multiple protocols. It is used primarily for storage area networks (SANs). The committee standardizing FC is the International Committee for Information Technology Standards (INCITS).

When configured as a Fibre Channel over Ethernet (FCoE)-FC gateway, the QFX3500 switch supports the transport of native FC traffic between FC switches and the gateway's native FC interfaces.

NOTE: Only the QFX3500 switch has native FC ports and supports native FC connection to the SAN. Only the QFX3500 can be configured as an FCoE-FC gateway, and only as a standalone switch or as a QFabric system Node device. FCoE-FC gateway configuration is not supported in Virtual Chassis or Virtual Chassis Fabric configurations.

FC concepts include:

- FC Fabrics on page 12
- FC Port Types on page 13
- FC Switches on page 13
- Adapters on page 13
- N_Port ID Virtualization (NPIV) on page 14
- FC Services on page 14

FC Fabrics

An FC fabric is a switched network topology that interconnects FC devices using FC switches, usually to create a SAN. An FC switch is a Layer 3 network switch that is compatible with the FC protocol, forwards FC traffic, and provides FC services to the components of the FC fabric. FC devices are usually servers or storage devices such as disk arrays.

Switches called FCoE forwarders (FCFs) perform a subset of FC switch functions. An FCF is a Layer 3 network switch that is compatible with the FC protocol and forwards FC traffic, but does not provide network services.

When configured as an FCoE-FC gateway, the QFX3500 switch acts as a proxy for the FCF functionality of an FC switch. The gateway provides FCoE devices on the Ethernet network access to the FC network without requiring the FC switches in the SAN to support Ethernet interfaces. The gateway is not an FCF and does not provide FC services.

FC network design often uses two fabrics (dual-rail topology) for redundancy. The two fabrics connect to edge devices but are otherwise unconnected, so that if one fabric goes down, the other fabric can continue to provide connectivity.
FC Port Types

The QFX3500 switch supports the following FC port types:

- **N_Port**—An N_Port is a port on the node of an FC device such as a server or a storage device and is also known as a node port.

- **F_Port**—An F_Port is a port on an FC switch that connects to an FC device N_Port in a point-to-point connection. F_Ports are also known as fabric ports.

These port types are a subset of the existing FC port types that can be supported in an FC fabric.

FC Switches

FC switches provide FC services to the FC network. FC switches forward Layer 3 traffic. They may transport a combination of native FC traffic and other traffic, such as Internet Small Computer Systems Interface (iSCSI) or FCoE, or they may transport only native FC traffic. When an FC switch supports FCoE, it combines FCoE termination functions with the FC stack on an FC switching element. This is also known as a dual-stack switch.

When FC switches support FCoE, they present virtual FC interfaces in the form of virtual F_Ports (VF_Ports) to the FCoE nodes (ENodes) on FCoE devices. A VF_Port is an endpoint in a virtual point-to-point connection with an ENode virtual N_Port (VN_Port). A VF_Port emulates a native FC F_Port and performs similar functions. A VF_Port is an intermediate port in a connection between an FCoE device such as a server in the Ethernet network and a storage device in the FC SAN.

FC switches that support FCoE contain at least one lossless Ethernet media access controller (MAC) paired with an FCoE controller. The lossless Ethernet MAC implements Ethernet extensions to avoid frame loss due to congestion. The FCoE controller instantiates and terminates virtual port instances as they are needed. Each VF_Port instance has one unique virtual link to an ENode VN_Port.

FCoE support also requires one FCoE Link End Point (LEP) for each VF_Port connection. An FCoE LEP is a virtual FC interface mapped onto the physical Ethernet interface. It transmits and receives FCoE frames on the virtual link, and handles FC frame encapsulation for traffic going from the FC switch to the FCoE device and frame de-encapsulation of traffic received from the FCoE device.

When you configure the QFX3500 switch as an FCoE-FC gateway, the gateway performs these FC-to-Ethernet and Ethernet-to-FC conversion functions so that the FC switch does not need Ethernet (FCoE) ports.

Adapters

FC host bus adapters (HBAs) in FC switches and devices perform functions similar to those of Ethernet adapters in Ethernet switches and devices. Switches that perform FCoE functions and FCoE devices have converged network adapters (CNAs) that support both native FC and Ethernet functionality.
**N_Port ID Virtualization (NPIV)**

FC requires a unique point-to-point link between the FC switch (F_Port) and each host N_Port. In order to avoid using one physical link for each F_Port to N_Port connection, the port connections must be virtualized so that they can share a physical link while maintaining logical separation.

FC accomplishes this by enabling you to create an independent virtual link for each FC session by mapping each session to a virtualized N_Port. This process is called N_Port ID virtualization (NPIV).

NPIV makes each virtual link look like a dedicated point-to-point link. In this way, multiple FC devices and multiple applications or virtual machines (VMs) on a single FC device can connect to an FC switch using one physical port instead of using a physical port for each connection. The virtual link creates a secure boundary between traffic from different sources on a single physical connection.

NPIV works by creating a unique virtual port identifier for each logical connection on a physical port. Conceptually, this is similar to splitting a single physical interface into multiple logical interfaces or subinterfaces. A virtual port identifier consists of the port’s unique worldwide name (WWN) combined with a Fibre Channel ID (FCID) that the FC switch assigns to the virtual connection. This creates a virtual host bus adapter (HBA) for each virtual link that uniquely identifies the link to the FC switch.

**FC Services**

When you configure the QFX3500 switch as an FCoE-FC gateway, the gateway connects FCoE devices in the Ethernet network to the FC fabric. The gateway does not provide FC services directly. The gateway logs in to the FC fabric and obtains FC services from the FC fabric, including:

- Management servers
  - Zone server— Defines which devices can connect to each other in the FC fabric.
  - Fabric configuration server—Discovers FC fabric topology and attributes.
  - Policy server—Distributes the rules for administering, managing, and controlling access to FC fabric resources.
  - HBA management server—Registers HBA information with the FC fabric.

- Domain manager—Allocates domain IDs to virtual switches.

- Fabric login server—Provides login services to the gateway so that the native FC ports on the gateway can perform initial fabric login (FLOGI) to the FC fabric and subsequent fabric discovery (FDISC) logins for the physical and virtual ports on the FCoE devices in the Ethernet network. This includes allocating Fibre Channel IDs (FCIDs) to ports.

- Name server—Discovers, registers, and unregisters N_Port attributes, including the attributes of the native FC ports on the gateway that connect to the FC fabric.

- Event server—Validates incoming events to ensure transaction integrity.
• Time server—Maintains a common time for devices in the FC fabric.

• Fabric controller
  • Fabric Shortest Path First (FSPF)—The FC fabric provides link-state path selection to the gateway.
  • State change notification (SCN) / registered state change notification server (RSCN)—Notifies the appropriate nodes when new devices come online, when other nodes fail, or when changes on an online node affect system operation.

Related Documentation
• Overview of Fibre Channel on page 4
• Understanding FCoE on page 20
• Understanding an FCoE-FC Gateway
• Understanding Fibre Channel Terminology on page 76
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 and EX4600 switch support 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.

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

- **Lossless Transport** on page 16
- **ETS** on page 17
- **DCBX** on page 18

**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 17
- **Buffer Management** on page 17
- **Physical Interfaces** on page 17

**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 150 meters (492 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-high priority flows. We recommend that you always configure a shaping rate to limit the amount of bandwidth a strict-high priority flow can consume by including the `shaping-rate` statement in the `[edit class-of-service schedulers]` hierarchy on the strict-high priority scheduler. This prevents a strict-high priority from starving other queues on the port.

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, ETS, and for Layer 2 and Layer 4 applications such as FCoE and iSCSI. DCBX is enabled or disabled on a per-interface basis.

**Related Documentation**

- Overview of Fibre Channel on page 4
- Understanding FCoE on page 20
- Understanding CoS Hierarchical Port Scheduling (ETS)
- Understanding CoS Flow Control (Ethernet PAUSE and PFC) on page 66
• Understanding DCBX on page 87
• Understanding Fibre Channel Terminology on page 76
• Example: Configuring CoS PFC for FCoE Traffic on page 115
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](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:** Mixed mode Virtual Chassis (VC) and Virtual Chassis Fabric (VCF) configurations do not support FCoE. Pure QFX5100 switch VCs and VCFs (consisting of only QFX5100 switches) support FCoE.

FCoE concepts include:

- FCoE Devices on page 21
- FCoE Frames on page 22
- Virtual Links on page 23
- FCoE VLANs on page 23
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 1 on page 22 shows a block diagram of the major 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
- Overview of Fibre Channel on page 4
- Understanding Fibre Channel on page 12
- Understanding DCB Features and Requirements on page 16
- Understanding FCoE Transit Switch Functionality on page 26
- Understanding VN_Port to VF_Port FIP Snooping on an FCoE Transit Switch on page 38
- Understanding FCoE LAGs
- Understanding CoS Flow Control (Ethernet PAUSE and PFC) on page 66
- Understanding Fibre Channel Terminology on page 76
- Configuring VLANs for FCoE Traffic on an FCoE Transit Switch on page 174
- Configuring an FCoE LAG
Understanding FCoE Transit Switch Functionality

You can use the switch 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 a switch acts as a transit switch, the VLANs you configure for FCoE traffic can use any of the switch ports 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.

QFX5100 and EX4600 switches use ELS software and 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. Do 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.

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.

NOTE: On a QFX3500 switch or on a QFabric system Node device, the same VLAN cannot be used in both transit switch mode and FCoE-FC gateway mode. (Only QFX3500 switches can be configured in FCoE-FC gateway mode.) If you configure both a transit switch and an FCoE-FC gateway on the same QFX3500 switch or QFabric system Node device, configure different FCoE VLANs for the transit switch and the FCoE-FC gateway.

Transit switch architecture differs from FCoE-FC gateway architecture. As an FCoE-FC gateway, the system 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. As a transit switch, the system forwards Ethernet traffic, and requires DCB configuration for lossless transport of that traffic and FIP snooping at FCoE device access ports, but not the FCoE-FC gateway features necessary for transporting FC traffic.

With the exception of mixed-mode Virtual Chassis (VC) and Virtual Chassis Fabric (VCF) configurations, switches support the DCB standards for ensuring lossless transport and low latency, and provide 10-Gbps ports for FCoE traffic. VC and VCF configurations that
use only QFX5100 switches support DCB standards. 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. The Technical Committee T11 organization specifications describe two types of FIP snooping:

- The FC-BB-5 specification describes VN_Port to VF_Port (VN2VF_Port) FIP snooping, which provides security for communication between FCoE device VN_Ports on the Ethernet network and FCF or FC switch VF_Ports.
- The FC-BB-6 specification describes VN_Port to VN_Port (VN2VN_Port) FIP snooping, which provides security for communication between FCoE device VN_Ports on the Ethernet 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 2 on page 28. The transit switch acts as a transparent DCB access layer between FCoE servers and the FC switch.

Figure 2: FCoE Transit Switch Connecting FCoE Devices to an FC Switch

The transit switch performs FIP snooping at the ports connected to the FCoE devices. For VN2VF_Port FIP snooping, at the SAN edge, the FC switch must be able to convert the FCoE traffic to native FC traffic. (VN2VN_Port FIP snooping switches traffic between
VN_Ports directly through the transit switch, without going through the FC switch, so no conversion of FCoE traffic to native FC traffic is needed.

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.

---

**Related Documentation**

- Overview of Fibre Channel on page 4
- Understanding DCB Features and Requirements on page 16
- Understanding an FCoE-FC Gateway
- Understanding FCoE on page 20
- Understanding FCoE LAGs
- Understanding VN_Port to VF_Port FIP Snooping on an FCoE Transit Switch on page 38
- Understanding VN_Port to VN_Port FIP Snooping on an FCoE Transit Switch on page 45
- Understanding Fibre Channel Terminology on page 76
- Configuring VLANs for FCoE Traffic on an FCoE Transit Switch on page 174
- Disabling Enhanced FIP Snooping Scaling
- Configuring an FCoE LAG
Understanding FCoE and FIP Session High Availability

High availability features maintain storage network sessions when a system process is terminated and during certain types of upgrades:

- High Availability for Fibre Channel Process Termination (FCoE-FC Gateway Mode) on page 30
- High Availability for FIP Snooping on page 30
- Nonstop Software Upgrade (QFabric Systems) on page 31

High Availability for Fibre Channel Process Termination (FCoE-FC Gateway Mode)

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. A session is a fabric login (FLOGI) or fabric discovery (FDISC) login to the FC SAN fabric, not an end-to-end server-to-storage session.

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.

High Availability for FIP Snooping

You can configure the system to perform FIP snooping on Ethernet interfaces that are connected to FCoE devices that have ENodes. The high availability function restores running FIP snooping sessions in case the Ethernet switching process is terminated.

The Ethernet switching process stores the FIP snooping state in a persistent storage module. If the Ethernet switching process terminates, the switch restores the existing...
FIP snooping sessions on the same interfaces that they were on before the Ethernet switching process terminated. The high availability features preserve:

- Logged in ENodes
- Discovered FCFs
- Existing sessions
- Existing FIP snooping filters

The complete restoration process, including reconciling all valid states, takes a maximum of 8 seconds. During the restoration process, the switch can learn a new FCF or a new FC switch, and new ENodes can log in to the FC network. However, FDISC messages from an ENode that is already logged in to the network might be dropped if the ENode has not yet been restored.

When the Ethernet switching process terminates ungracefully, the FIP keepalive timer is reset to the normal initial value, not the value at the time of the Ethernet switching process termination.

In the event of an Ethernet switching process termination, ENodes remain logged in, and existing sessions are not interrupted.

NOTE: An Ethernet switching process restart and session restoration resets the FIP snooping statistics.

**Nonstop Software Upgrade (QFabric Systems)**

On QFabric system Node groups that have more than one Node device, nonstop software upgrade (NSSU) enables you to upgrade the Node devices with minimal packet loss and maximum uptime. NSSU automates software upgrades on the QFabric system components in an orderly and consistent manner to maximize system uptime.

The system upgrades components with redundant architectures, such as redundant server Node groups and network Node groups that have two or more members, in stages. While the system upgrades one component, the redundant component continues to function.

For example, while one member of a redundant server Node group is upgraded, the other member continues to forward traffic. When the first Node group member completes the upgrade, it comes online while the system upgrades the second member.

NSSU provides high availability for the lossless traffic forwarding required to support storage networks. If your system design includes redundancy (redundant Node devices in Node groups, LAGs, and so on) so that an alternate traffic path is available, when you upgrade a Node device, traffic is not impacted.

In fully redundant topologies, NSSU preserves FIP session, FIP snooping filter, VN2VF_Port session, and VN2VN_Port session information and prevents traffic loss in most cases.
An exception is that Node devices that are directly connected to ENodes experience momentary traffic loss when the Node device reboots.

### Related Documentation
- Understanding an FCoE-FC Gateway
- Understanding FCoE on page 20
- Understanding Nonstop Software Upgrade for QFabric Systems
- Performing a Nonstop Software Upgrade on the QFabric System

## Understanding FIP Functions

Fibre Channel over Ethernet (FCoE) Initialization Protocol (FIP) performs four major functions:

- **FIP VLAN discovery**: FCoE device FCoE nodes (ENodes) discover the FCoE VLANs on which to transmit and receive FIP and FCoE traffic.
- **FIP discovery**: FCoE devices discover Fibre Channel (FC) switches to which they can connect.
- **Initialization**: FCoE devices perform fabric login (FLOGI) and fabric discovery (FDISC) to create a virtual link with an FC switch.
- **Maintenance**: The switch ensures that the virtual link between the FCoE device and the FC switch remains valid, and also that the link termination logout (LOGO) functions properly.

When you configure the switch as an FCoE-FC gateway (QFX3500 switch only, as a standalone switch or as a QFabric system Node device), it converts FIP requests and information from FCoE devices into FC requests and information and relays them to the FC switch. To FCoE devices, the gateway appears to be an FCoE forwarder (FCF) and presents virtual fabric port (VF_Port) interfaces to the server ENode. To FC switches, the gateway appears to be an FC device that supports N_Port ID virtualization (NPIV) and presents an N_Port interface to the FC switch F_Port interface. When you configure the switch as an FCoE transit switch, you do not configure FIP parameters on the switch.

FIP FLOGI, FDISC, and LOGO are similar to the same processes in the native FC protocol.

This topic describes:

- FIP VLAN Discovery on page 33
- FIP Discovery on page 33
- FIP FLOGI on page 34
- FIP FDISC on page 35
- FIP Maintenance (Keepalive Messages) on page 35
- FIP LOGO on page 36
FIP VLAN Discovery

The gateway supports FIP VLAN discovery. Host ENodes use FIP VLAN discovery to discover the FCoE VLANs on which they will send and receive FIP and FCoE traffic and on which they will establish a virtual link with the FC switch. This means FCoE devices do not need manually configured FCoE VLANs.

FIP VLAN discovery and notification takes place on the native VLAN that the FCoE device uses for Ethernet traffic:

1. The ENode sends a FIP VLAN discovery request to a multicast address called \textit{ALL-FCF-MACs} to which all FC switches and FCFs on the VLAN listen.
2. The FC switches and FCFs respond on the native VLAN with a list of the FCoE VLANs that are available for login.
3. The ENode selects an FCoE VLAN and continues the FIP process on that VLAN.

Except for FIP VLAN discovery, all other FIP and FCoE traffic runs on an FCoE VLAN.

\textbf{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.

FIP Discovery

The FIP discovery process allows an FCoE device ENode MAC to locate (discover) the FC switches in the FCoE VLAN to which it belongs. The ENode selects an FC switch to log in to from the available FC switches. Either the ENode MAC or the FC switch can initiate the FIP discovery process.

Server ENode MACs initiate FIP discovery:

1. When an ENode MAC comes online, it sends a multicast discovery solicitation message on its FCoE VLAN to a multicast address called \textit{ALL-FCF-MACs} to which all FCFs (including the FCF functionality of FC switches) on the VLAN listen. The discovery solicitation message includes the ENode's addressing mode and the maximum protocol data unit (PDU) size the ENode MAC uses for FCoE traffic.

   The ENode uses the globally unique ENode MAC address assigned to it by the converged network adapter (CNA) manufacturer as an identifier in the FIP frame header.

2. The FCFs on the VLAN that have a similar supported addressing mode, match the maximum FCoE size, and can accept a login from the ENode reply to the discovery solicitation message by sending a solicited unicast discovery advertisement message to the soliciting ENode MAC.

3. The ENode MAC compiles a list of FCFs that are available for login, selects an FCF (the FCF with the highest priority setting), and is then ready to log in to the FCF.
The FIP discovery process is similar when the FC switch or FCF initiates discovery:

1. FCF MACs periodically send unsolicited multicast discovery advertisements on the FCoE VLAN to the ALL-ENode-MACs multicast address, to which all ENode MACs on the VLAN listen. The FIP keepalive advertisement period timer (FKA_ADV_PERIOD) controls the interval between multicast discovery advertisements. The multicast discovery advertisements inform ENodes on the VLAN that FCF VF_Ports are available for establishing virtual links with ENode VN_Ports.

2. ENodes on the FCoE VLAN create an entry for the FCF-MAC in their FCF-MAC lists.

3. An ENode can respond to the unsolicited multicast discovery advertisement with a unicast discovery solicitation message to the FCF.

4. Upon receiving the ENode's unicast discovery solicitation, the FCF replies with a unicast discovery advertisement sent to the ENode MAC.

After the ENode MAC selects an FCF to log in to, FIP initialization begins. To proceed from discovery to initialization, the server ENode addressing mode must match the FCF addressing mode and maximum FCoE size. In addition, the FCF must be configured to allow FIP FLOGI from that ENode.

**FIP FLOGI**

FIP initialization is the server ENode login process to the FCF after the ENode discovers the FCFs (including FC switches) on the FCoE VLAN:

1. The ENode sends a fabric login (FLOGI) request message to the FCF.

2. The FCF replies to confirm the ENode login and provides the ENode a locally unique MAC address to use for FCoE frame transactions. The locally unique MAC address identifies the VN_Port interface of the ENode for the session the login establishes. (The ENode continues to use the globally unique ENode MAC address for FIP frame transactions.)

The locally unique ENode MAC address for FCoE operations depends on whether the ENode address mode is configured as a fabric-provided MAC address (FPMA) or as a server-provided MAC address (SPMA; the gateway does not support ENodes in SPMA mode and rejects login attempts from ENodes in SPMA mode):

- For FPMA mode, the FCF provides a MAC address to the ENode during the FIP FLOGI exchange. The FPMA MAC address is a 48-bit value that is unique to the local fabric and consists of a 24-bit FCoE mapped address prefix (FC-MAP) and a 24-bit FC identifier (FCID). You can configure the FC-MAP value on the FCF or use the default value of 0EFC00h. The FCoE device must use the same FC-MAP value as the FCF, or else discovery and login fail.

- For SPMA mode, the server provides its MAC address to the FCF. The FCF compares the server MAC address to a list of addresses approved for FCoE access. The gateway does not support ENodes in SPMA mode.

Successful login instantiates a secure virtual link between the ENode and the FCF and terminates the FIP virtual link instantiation phase. The initiating server behind the ENode
can exchange FC payloads with storage devices in the FC SAN by sending FCoE frames over the virtual link.

**FIP FDISC**

After an ENode successfully logs in to an FCF and establishes a virtual link, the ENode can request more virtual links (sessions) over the same physical link by sending a FIP fabric discovery (FDISC) request. FDISC allows the creation of multiple separate secure VN_Port virtual links on one physical link. Each virtual link receives a locally unique identifier from the FCF to enable security and separation between the VN_Port virtual links sharing a physical ENode port. This is called N_Port ID virtualization (NPIV).

FDISC is similar to FLOGI in that it requests a login and a unique ID from the FCF. The difference is that FLOGI obtains the initial login and ID for the physical link, whereas FDISC obtains additional logins and IDs so that multiple virtual links can share one physical link securely.

After a VN_Port FDISC is complete, the application using that VN_Port can send FCoE frames over the virtual link.

**FIP Maintenance (Keepalive Messages)**

Although FCoE protocol handles the payload communication between the initiating ENode and the target FC device, FIP continues to run in the background. FIP constantly updates ENode FCF lists by listening to the periodic FCF multicast discovery advertisements, and it verifies the ability to reach the FCF by transmitting periodic FIP keepalive advertisements.

The ENode sends periodic ENode FIP keepalive advertisements to the FCF with the ENode MAC address as the identifier. The ENode also sends periodic VN_Port FIP keepalive advertisements on behalf of each VN_Port on the ENode, using the VN_Port MAC address as the source MAC. The VN_Port FIP keepalive advertisements occur every 90 seconds. The keepalive advertisements reset the session timer for the virtual link connection to the FCF. If the FCF does not receive a keepalive advertisement for a logged-in ENode or VN_Port before the session timer expires, the virtual link is terminated.

The periodic unsolicited multicast discovery advertisements the FCF sends to the ALL-ENode-MACs address continuously verify that the FCF is still reachable. The ENode and the FCF periodic unsolicited multicast discovery advertisements occur at the configured FIP keepalive advertisement period interval (FKA_ADV_PERIOD) plus or minus a random offset to prevent a flood of simultaneous keepalive advertisements.

If the FCF does not receive the ENode keepalive advertisements before the FCF's FIP keepalive timer expires, the FCF considers the virtual link to the ENode as "down" and terminates the virtual link to the ENode. The keepalive timer expires in 2.5 times the configured timer value. This also terminates any VN_Port virtual links instantiated by that ENode.

If the FCF does not receive a VN_Port keepalive advertisement before the FCF's FIP keepalive timer expires, the FCF considers the virtual link to the VN_Port as "down" and terminates the virtual link to that VN_Port. The VN_Port keepalive timer expires in 2.5 times the configured timer value.
If the ENode does not receive the FCF unsolicited multicast discovery advertisement before the ENode's FIP keepalive timer expires, the ENode considers the virtual link to the FCF as "down" and all of the VN_Port virtual links to that FCF on the ENode are terminated.

**FIP LOGO**

FIP handles ENode and VN_Port logout when a session is finished.

**Related Documentation**

- Overview of FIP on page 9
- Understanding FIP Implementation on an FCoE-FC Gateway
- Understanding FIP Parameters on an FCoE-FC Gateway
- Understanding Fibre Channel Virtual Links on page 36
- Understanding FCoE on page 20

**Understanding Fibre Channel Virtual Links**

A virtual link emulates a secure point-to-point connection between the virtual node port (VN_Port) of a Fibre Channel over Ethernet (FCoE) node (ENode) and the virtual fabric port (VF_Port) of an FCoE forwarder (FCF). The combination of the FCF media access control (MAC) address and the VN_Port MAC address uniquely identifies each virtual link. Uniquely identifying each virtual link enables the logical separation of traffic that belongs to each virtual link. A single physical link between an ENode and an FCF can carry multiple virtual links and maintain secure, separate transport of traffic on the different virtual links.

Virtual links are necessary because Fibre Channel protocol does not recognize multipoint-to-point connections. Even when multiple connections are aggregated on one physical port, FCoE Initialization Protocol (FIP) presents each virtual link as an individual point-to-point link between an ENode VN_Name and an FCF VF_Name.

**Related Documentation**

- Overview of FIP on page 9
- Understanding FIP Functions on page 32
- Understanding FIP Implementation on an FCoE-FC Gateway
- Understanding FIP Parameters on an FCoE-FC Gateway
- Understanding FCoE on page 20
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:
- Understanding OxID Hash Control for FCoE Traffic Load Balancing on QFabric Systems
- Enabling and Disabling CoS OxID Hash Control on Standalone Switches on page 171

Copyright © 2014, Juniper Networks, Inc.
Understanding VN_Port to VF_Port FIP Snooping on an FCoE Transit Switch

Fibre Channel over Ethernet (FCoE) Initialization Protocol (FIP) snooping is a security mechanism that is designed to prevent unauthorized access and data transmission to a Fibre Channel (FC) network. It works by filtering traffic to permit only servers that have logged in to an FC network to access that network.

You explicitly enable VN_Port to VF_Port (VN2VF_Port) FIP snooping (FC-BB-5) on FCoE VLANs when the switch is an FCoE transit switch at the access edge that connects FCoE devices on the Ethernet network to FC switches or gateways at the FC storage area network (SAN) edge. The transit switch applies FIP snooping filters at the ports associated with the FCoE VLANs on which you enable VN2VF_Port FIP snooping. An FCoE transit switch is a data center bridging (DCB) switch with FIP snooping capability.

An FCoE device that has a converged network adapter (CNA) uses the FIP process to log in to the FC network as an FCoE Node (ENode). The login process establishes a dedicated virtual link between a virtual N_Port (VN_Port) on the ENode and a virtual F_Port (VF_Port) on the FC switch. This dedicated virtual link emulates a point-to-point connection. The emulated connection is called a virtual link.

Virtual links pass transparently through the transit switch. The ENode VN_Port and the FC switch VF_Port do not detect the transit switch, and virtual links appear to be direct point-to-point links.

The switch applies VN2VF_Port FIP snooping firewall filters at the FCoE-network facing ports associated with the FCoE VLANs on which you enable VN2VF_Port FIP snooping. FIP snooping provides security for virtual links by creating firewall filters based on information gathered (snooped) about FC devices during FIP transactions.

The switch also supports VN_Port to VN_Port (VN2VN_Port) FIP snooping (FC-BB-6) to allow FCoE initiators and targets to communicate directly through the switch without going through an FCoE forwarder (FCF) or an FC switch, as described in "Understanding VN_Port to VN_Port FIP Snooping on an FCoE Transit Switch" on page 45.

NOTE: 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 for VN2VF_Port FIP snooping traffic and others for VN2VN_Port FIP snooping traffic. On FCoE VLANs that are configured as VN2VN_Port snooping VLANs, VN2VF_Port FIP snooping traffic is dropped.

When you enable VN2VF_Port FIP snooping on an FCoE VLAN, the system snoops VN_Port to VF_Port packets and enforces security only on VN2VF_Port virtual links.

When you enable VN2VN_Port FIP snooping on an FCoE VLAN, the system snoops VN_Port to VN_Port packets and enforces security only on VN2VN_Port virtual links.
This topic describes:

- FC Network Security on page 39
- VN2VF_Port FIP Snooping Functions on page 40
- FIP Snooping Firewall Filters on page 40
- FIP Snooping Session Scalability on page 40
- VN2VF_Port FIP Snooping Implementation on page 41
- T11 VN2VF_Port FIP Snooping Specification on page 44

**FC Network Security**

In traditional FC networks, the FC switch is usually a trusted entity, and server ENodes connect directly to its VF_Ports. After an ENode gains access to the network through the fabric login (FLOGI) process, the FC switch enforces zoning configurations, ensures that the ENode uses valid addresses, monitors the connection, and performs other security functions to prevent unauthorized access.

However, FCoE exposes FC frames to Ethernet networks, which do not have the same level of security as native FC networks. VN2VF_Port FIP snooping firewall filters emulate the native FC network security functions by preventing unauthorized access to the FC switch through the transit switch and by ensuring the security of the virtual link between each ENode and the FC switch, as shown in Figure 3 on page 39. VN2VF_Port FIP snooping also prevents man-in-the-middle attacks.

**Figure 3: FCoE Transit Switch Performs VN2VF_Port FIP Snooping**

The transit switch performs VN2VF_Port 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.
VN2VF_Port FIP Snooping Functions

When VN2VF_Port FIP snooping is enabled, the transit switch sets and applies filters to block all FCoE traffic by default. The transit switch monitors FIP logins, solicitations, and advertisements that pass through it and gathers information about the ENode address and the address of the port on the FC switch. The transit switch uses the information to construct firewall filters that permit access only to logged-in ENodes. All other traffic on the VLAN is denied.

For example, when an ENode on an FCoE VLAN performs a successful login to an FC switch port, the transit switch snoops the FIP information and constructs a firewall filter that provides access for the ENode to that port on the FC switch.

The firewall filters enable FCoE frames to pass through the transit switch only on a virtual link established between an FCoE device ENode VN_Port and the FC switch VF_Port to which it has logged in. The firewall filters ensure that ENodes can only connect to the FC switches they have successfully logged in to and that only valid FCoE traffic along valid paths is transmitted. VN2VF_Port FIP snooping maintains the filters by tracking FCoE sessions (ENode to FCF sessions).

FIP Snooping Firewall Filters

The effect of the firewall filters is to protect the FCoE ports. VN2VF_Port FIP snooping performs the following actions and checks to ensure that FCoE traffic is valid:

- Denies ENodes that use the FC switch media access control (MAC) address as the source address.
- Enables ENodes to transmit FIP and FCoE frames to the FC switch address.
- Ensures that the FCoE source address the FC switch assigns or accepts is only used for FCoE traffic.
- Ensures that FCoE frames are only addressed to the accepting FC switch.

FIP Snooping Session Scalability

Enhanced FIP snooping session scaling, which supports up to 2,500 sessions, is enabled by default. On QFabric systems, if you want to disable enhanced FIP snooping scaling (which reduces the number of supported sessions to 376 sessions), you can do so as described in Disabling Enhanced FIP Snooping Scaling.

By default, up to 2500 total FIP snooping sessions are supported on an interface, an FCoE-FC gateway fabric (only supported on QFX3500 switches configured as standalone switches or as QFabric system Node devices), a switch, a QFabric Node device, or a QFabric Node group. For example, you can:

- Place all 2500 sessions on one FCoE interface.
- Split the 2500 sessions among multiple FCoE interfaces on one FCoE VLAN.
- Split the 2500 sessions among multiple FCoE interfaces on multiple FCoE VLANs.
• Split the 2500 sessions among the FCoE interfaces on multiple gateway FC fabrics on a switch.

• Split the 2500 sessions among the FCoE interfaces on multiple gateway FC fabrics on multiple Node devices in a QFabric Node group.

Regardless of how you allocate the sessions among interfaces and local FC fabrics on a switch or on a QFabric system Node device or Node group, the combined FIP session limit is a maximum of 2500 sessions.

NOTE: The total number of sessions the system can support is the combined number of VN2VF_Port sessions and VN2VN_Port sessions. If VN2VN_Port sessions are active, the total number of available VN2VF_Port sessions is reduced.

VN2VF_Port FIP Snooping Implementation

You enable VN2VF_Port FIP snooping on a per-VLAN basis on VLANs that carry FCoE traffic. The switch snoops FIP frames at the ports associated with FCoE VLANs enabled for VN2VF_Port FIP snooping. The switch then installs the resulting firewall filters on the ports to ensure that all VN2VF_Port FIP snooping occurs on the switch network edge.

VN2VF_Port FIP snooping FCoE VLANs must meet the following criteria:

• An FCoE VLAN should be dedicated to FCoE traffic only.

• An FCoE VLAN cannot support both VN2VF_Port FIP snooping and VN2VN_Port FIP snooping simultaneously. You must configure separate FCoE VLANs for VN2VF_Port FIP snooping traffic and for VN2VN_Port FIP snooping traffic.

NOTE: Changing an FCoE VLAN from VN2VF_Port FIP snooping mode to VN2VN_Port snooping mode terminates the existing virtual links on the VLAN. The transit switch removes the existing FIP snooping filters, creates the new FIP snooping filters, and applies them to the FIP snooping ports. If you downgrade the software to Junos OS Release 12.1 or earlier, VLANs configured for VN2VN_Port FIP snooping revert to VN2VF_Port FIP snooping VLANs.

• For systems that use software that does not support Enhanced Layer 2 Software (ELS) CLI, configure all access ports that belong to an FCoE VLAN (ports connected to a converged network adapter [CNA] in an FCoE device) in tagged-access port mode. Access ports associated with an FCoE VLAN should not be configured as access ports or trunk ports on these platforms, although trunk port configuration is supported. However, on switches that use the ELS CLI, configure access ports that belong to an FCoE VLAN in trunk interface mode.

• All ports connected to an FC switch (or FCoE forwarder) must be configured in trunk port mode. Ports connected to an FC switch must be configured as trusted ports.
• FIP traffic uses the native VLAN (FIP VLAN discovery and notification frames are exchanged as untagged packets).
• All FCoE VLAN traffic must be tagged and cannot belong to the native VLAN.
• FCoE VLAN traffic cannot be untagged or priority-tagged.

When you enable VN2VF_Port FIP snooping, the switch inspects FIP frames.

The VN2VF_Port FIP snooping implementation includes these considerations:

• ENode-Facing Interfaces on page 42
• Network-Facing Interfaces on page 43
• FC-MAP on page 43

**ENode-Facing Interfaces**

When the interfaces that belong to an FCoE VLAN connect directly to FCoE devices (there is no other transit switch between the FCoE devices and the switch), we recommend that you enable VN2VF_Port FIP snooping on all FCoE VLANs that connect VN_Ports to VF_Ports. Enabling FIP snooping ensures secure connections between server ENodes and FC switches. (Enabling VN2VN_Port FIP snooping ensures secure connections on FCoE VLANs that connect VN_Ports to other VN_Ports). FIP snooping should always be enabled at the access edge.

Systems that run Enhanced Layer 2 Software (ELS) support a slightly different configuration on ENode-facing interfaces than systems that do not run ELS. This section describes:

• Non-ELS Port Mode for FCoE Interfaces on page 42
• ELS Interface Mode for FCoE Interfaces on page 43
• Trusted and Untrusted FCoE Interfaces on page 43

**Non-ELS Port Mode for FCoE Interfaces**

The interfaces that belong to FCoE VLANs (interfaces that connect to CNAs in FCoE devices) on systems that do not support ELS should be configured in **tagged-access** port mode. After you enable VN2VF_Port FIP snooping on an FCoE VLAN, the transit switch denies FCoE traffic from any ENode on that VLAN until the ENode performs a valid fabric login with an FC switch.

The **tagged-access** port mode was not available in Junos OS Release 11.3 and prior releases. In Release 11.3 and earlier, **trunk** port mode was used for Ethernet interfaces that connected to FCoE access devices. Because **tagged-access** mode is now available, using **trunk** mode for interfaces connected to FCoE CNAs is not recommended.

If an existing configuration uses **trunk** mode for ports connected to FCoE CNAs, you can change the port mode to **tagged-access** without disrupting traffic. Although we recommend changing the port mode of these ports from **trunk** to **tagged-access** as a best practice, it is not mandatory. New configurations should use **tagged-access** mode for interfaces that connect to FCoE devices.
**ELS Interface Mode for FCoE Interfaces**

The interfaces that belong to FCoE VLANs (interfaces that connect to CNAs in FCoE devices) on systems that support ELS should be configured in **trunk** interface mode. After you enable VN2VF_Port FIP snooping on an FCoE VLAN, the transit switch denies FCoE traffic from any ENode on that VLAN until the ENode performs a valid fabric login with an FC switch.

**Trusted and Untrusted FCoE Interfaces**

Do not configure ENode-facing interfaces as FCoE trusted interfaces when VN2VF_Port FIP snooping is enabled on those interfaces. If you enable VN2VF_Port FIP snooping on an FCoE VLAN and you configure ENode-facing interfaces that are members of the FIP snooping VLAN as `fcoe-trusted`, then FCoE devices might not be able to log in to the FC network.

Changing ports from untrusted to trusted removes any existing VN2VF_Port FIP snooping filters from the ports and terminates the existing sessions. Changing the fabric ports from trusted to untrusted forces all of the FCoE sessions on those ports to log out so that when the ENodes and VN_Ports log in again, the switch can build the appropriate VN2VF_Port FIP snooping filters.

**Network-Facing Interfaces**

When the switch acts as an FCoE transit switch, you must configure any interface that is connected to a switch as an FCoE trusted interface in **trunk** port mode and as a 10-Gigabit Ethernet interface.

Switch-facing Ethernet interfaces have the following requirements and behaviors:

- You must explicitly configure switch-facing trunk ports on an FCoE transit switch as FCoE trusted interfaces.
- After you configure an FC switch-facing trunk port as a trusted interface, the FCoE transit switch always processes FC switch frames because they come from a source on a trusted interface.
- All ports in an FCoE VLAN must be configured as tagged access or trunk ports.

**FC-MAP**

When the switch acts as an FCoE transit switch and you enable VN2VF_Port FIP snooping on an FCoE VLAN, you can optionally specify a 24-bit FCoE mapped address prefix (FC-MAP) value. On a given VLAN, the transit switch learns only those FC switches that have a matching FC-MAP value. If the transit switch FCoE VLAN FC-MAP value does not match the FC switch FC-MAP value, the transit switch does not discover the FC switch on that VLAN, and the ENodes on that VLAN cannot access the FC switch. An FCoE VLAN can have one and only one FC-MAP value.

The FC-MAP value is a MAC address prefix unique to an FC switch in the FC SAN fabric that the FC switch uses to identify FCoE traffic for a given FC fabric (traffic on a particular FCoE VLAN). The FC switch combines the FC-MAP value with a unique 24-bit FCID value for the ENode VN_Port during the login process. This creates a 48-bit identifier that is
unique to the fabric. The FC switch assigns this 48-bit value to the ENode VN_Port as its MAC address and unique identifier for the session. Each VN_Port session the ENode establishes with the FC switch receives a unique FCID from the FC switch, so an FCoE device can host multiple virtual links (one for each VN_Port) to an FC switch, each with a 48-bit MAC address that is unique to the fabric.

The VN2VF_Port FIP snooping filter compares the configured FC-MAP value with the FC-MAP value in the header of frames coming from the ENode VN_Port. If the values do not match, the transit switch denies access.

NOTE: Changing the FC-MAP value causes all logins to be dropped and forces ENodes to log in again.

NOTE: Do not configure static MAC addresses with the FC-MAP value as a prefix (the first 24 bits of the MAC address). If you configure a static MAC address that uses the FC-MAP value as a prefix, the system deletes the static MAC address automatically after you enable FIP snooping. The static MAC address configuration is not restored even if you disable FIP snooping later. (The system considers a static MAC address with the FC-MAP value as the prefix to be a misconfiguration.) Do not use a MAC address with the FC-MAP value as the prefix for any traffic other than the FIP snooping traffic when the switch is acting as a transit switch.

T11 VN2VF_Port FIP Snooping Specification


Related Documentation

- Overview of Fibre Channel on page 4
- Understanding DCB Features and Requirements on page 16
- Understanding FCoE Transit Switch Functionality on page 26
- Understanding an FCoE-FC Gateway
- Overview of FIP on page 9
- Understanding VN_Port to VN_Port FIP Snooping on an FCoE Transit Switch on page 45
- Understanding FCoE LAGs
- Understanding FIP Snooping, FBF, and MVR Filter Scalability on page 53
- Configuring VN2VF_Port FIP Snooping and FCoE Trusted Interfaces on an FCoE Transit Switch on page 179
- Disabling VN2VF_Port FIP Snooping on an FCoE-FC Gateway Switch Interface
- Disabling Enhanced FIP Snooping Scaling
Understanding VN_Port to VN_Port FIP Snooping on an FCoE Transit Switch

VN_Port to VN_Port (VN2VN_Port) Fibre Channel over Ethernet (FCoE) Initialization Protocol (FIP) snooping (FC-BB-6) on an FCoE transit switch is conceptually similar to VN_Port to VF_Port (VN2VF_Port) FIP snooping (FC-BB-5) on an FCoE transit switch. An FCoE transit switch is a data center bridging (DCB) switch with FIP snooping capability. VN2VN_Port FIP snooping provides security in the form of filters. The filters help prevent unauthorized access and data transmission on a bridge that connects ENodes on the Ethernet network.

The main difference between VN2VN_Port FIP snooping and VN2VF_Port FIP snooping is that you use VN2VN_Port FIP snooping when the FCoE devices reside on the Ethernet network, so there is no need to forward traffic between FCoE devices to the Fibre Channel (FC) network, and you use VN2VF_Port FIP snooping when FCoE devices on the Ethernet network need to access targets on the FC network, so FCoE traffic must be forwarded to the FC network. See “Understanding VN_Port to VF_Port FIP Snooping on an FCoE Transit Switch” on page 38 for information about VN2VF_Port FIP snooping.

You enable VN2VN_Port FIP snooping on the FCoE VLAN that transports the VN2VN traffic. The transit switch applies VN2VN_Port FIP snooping filters at the ports associated with the FCoE VLANs on which you enable VN2VN FIP snooping.

A key benefit of VN2VN_Port FIP snooping is that it enables FCoE initiators and targets to communicate directly through the switch without going through an FCoE forwarder (FCF) or an FC switch. The transit switch does not differentiate between initiators and targets because the transit switch sees both VN_Ports as FIP virtual link end points. Direct VN2VN_Port communication requires secure access (FIP snooping filters) because ENodes are not trusted entities.

This topic describes:

- VN2VN_Port FIP Snooping and FIP Snooping Virtual Links on page 46
- VN2VN_Port Communication Modes on page 46
- Network Security on page 47
- VN2VN_Port FIP Snooping Functions on page 47
- Scalability on page 47
- VN2VN_Port FIP Snooping Implementation on page 48
- ENode-Facing Interfaces on page 48
- Network-Facing Interfaces (Connecting to Another Transit Switch) on page 50
- Beacon Period (VN2VN_Port FIP Snooping Link Maintenance) on page 50
- QFabric System Differences in VN2VN_Port FIP Snooping Traffic Handling on page 50
VN2VN_Port FIP Snooping and FIP Snooping Virtual Links

FIP snooping under the T11 FC-BB-5 specification requires that an FC switch or an FCF be in the path between two VN_Ports when they communicate. Introduced in the T11 FC-BB-6 specification (see http://www.t11.org/ftp/t11/pub/fc/bb-6/10-019v3.pdf), VN2VN_Port FIP snooping allows the FCoE transit switch to connect two VN_Ports to each other directly, without going through an FC switch or an FCF, provided that the ENodes have logged in to the FC network.

In VN2VF_Port FIP snooping, when an ENode logs in to the FC network, the FCoE transit switch snoops the FIP communication between the ENode and the FC switch. In VN2VN_Port FIP snooping mode, the transit switch creates filters on the switch access ports to control VN_Port access to other VN_Ports on the Ethernet network. The VN2VN_Port FIP snooping filters allow the switch to establish a dedicated virtual link that emulates a point-to-point connection between two VN_Ports, through the switch.

Virtual links pass transparently through the transit switch. The VN_Ports do not detect the transit switch, and virtual links appear to be direct point-to-point links.

You explicitly enable VN2VN_Port FIP snooping on FCoE VLANs when the switch or QFabric system Node device is an FCoE transit switch connecting FCoE devices on the Ethernet network to each other and to FC switches or gateways at the FC storage area network (SAN) edge.

NOTE: An FCoE VLAN can support either VN2VF_Port FIP snooping or VN2VN_Port FIP snooping, but not both. Configure separate FCoE VLANs for VN2VF_Port FIP snooping traffic and for VN2VN_Port FIP snooping traffic. On FCoE VLANs that are configured as VN2VN_Port FIP snooping VLANs, VN_Port to VF_Port traffic is dropped.

When you enable FIP snooping, the system snoops VN2VF_Port packets and enforces security only on VN_Port to VF_Port virtual links. When you enable VN2VN_Port FIP snooping, the system snoops VN_Port to VN_Port FIP packets and enforces security only on VN_Port to VN_Port virtual links.

The transit switch applies VN2VN_Port FIP snooping filters at the ports associated with the FCoE VLANs on which you enable VN2VN_Port FIP snooping. VN2VN_Port FIP snooping provides security for virtual links by creating filters based on information gathered (snooped) about FCoE devices during FIP transactions.

VN2VN_Port Communication Modes

The transit switch supports two VN2VN_Port communication modes:

- Point-to-point mode
- Multipoint mode
In point-to-point mode, two ENodes are connected to the network and form a single VN_Port to VN_Port virtual link. This is analogous to the point-to-point FC link between an FC initiator and an FC target.

In multipoint mode, multiple ENodes are connected to the network and form multiple virtual links. Each virtual link is created between one pair of VN_Ports. This is analogous to loop mode in traditional FC networks.

The VN2VN_Port communication mode is not configured; it is determined by the number of ENodes connected to the network.

**Network Security**

In traditional FC networks, the FC switch is usually a trusted entity and the server ENodes are untrusted entities. The ENodes connect directly to the FC switch VF_Ports. After an ENode gains access to the network through the fabric login (FLOGI) process, the FC switch enforces zoning configurations, ensures that the ENode uses valid addresses, monitors the connection, and performs other security functions to prevent unauthorized access.

However, FCoE exposes FC frames to Ethernet networks, which do not have the same level of security as native FC networks. VN2VN_Port FIP snooping filters emulate the native FC network security functions by preventing unauthorized access and by ensuring the security of the virtual link between ENode VN_Ports. The transit switch performs VN2VN_Port FIP snooping at the ports connected to the FCoE VN_Port devices.

**VN2VN_Port FIP Snooping Functions**

When you enable VN2VN_Port FIP snooping, the transit switch sets and applies filters to block all FCoE traffic on the VLAN by default. The transit switch monitors FIP logins, solicitations, and advertisements that pass through it and gathers information about the ENode address. The transit switch uses the information to construct filters that permit access only to logged-in ENodes. All other traffic on the VLAN is denied.

The filters enable FCoE frames to pass through the transit switch only on a virtual link established between two VN_Ports. The filters ensure that ENodes can only connect to other ENodes if they have successfully logged in to each other, and that only valid FCoE traffic along valid paths is transmitted. VN2VN_Port FIP snooping maintains the filters by tracking VN_Port to VN_Port sessions.

**Scalability**

Because ENodes are untrusted and the system needs to apply filters to untrusted FIP snooping interfaces, the total number of combined VN2VN_Port FIP snooping sessions per switch is 376 sessions (ENode to ENode sessions) on untrusted interfaces. On interfaces that are configured as trusted interfaces, no FIP snooping filters are applied.

---

**NOTE:** The total number of sessions the system can support is the combined number of VN2VF_Port sessions and VN2VN_Port sessions. If VN2VF_Port sessions are active, the total number of available VN2VN_Port sessions is reduced.
VN2VN_Port FIP Snooping Implementation

You enable VN2VN_Port FIP snooping on a per-VLAN basis on VLANs that carry FCoE traffic. The switch snoops FIP frames at the ports associated with FCoE VLANs enabled for VN2VN_Port FIP snooping. The switch then installs the resulting filters on the ENode-facing ports to ensure that all FIP snooping occurs on the switch network edge.

VN2VN_Port FIP snooping FCoE VLANs must meet the following criteria:

- An FCoE VLAN should be dedicated to FCoE traffic only.
- An FCoE VLAN cannot support both VN2VF_Port FIP snooping (FC-BB-5) and VN2VN_Port FIP snooping (FC-BB-6) simultaneously. You must configure separate FCoE VLANs for FIP snooping traffic and for VN2VN_Port FIP snooping traffic.

**NOTE:** Changing an FCoE VLAN from VN2VF_Port FIP snooping mode to VN2VN_Port FIP snooping mode terminates the existing virtual links on the VLAN. The transit switch removes the existing FIP snooping filters, creates the new FIP snooping filters, and applies them to the FIP snooping ports. If you downgrade the software to Junos OS Release 12.1 or earlier, VLANs configured for VN2VN_Port FIP snooping revert to VN2VF_Port FIP snooping VLANs.

- For switches that do not run Enhanced Layer 2 Software (ELS), as a best practice, you should configure all access ports that belong to an FCoE VLAN (ports connected to a converged network adapter [CNA] in an FCoE device) in **tagged-access** port mode. However, access and trunk port modes are also supported. For switches that use ELS, configure access ports that belong to an FCoE VLAN in **trunk** interface mode.
- Access ports should be configured as untrusted ports.
- All ports connected to another transit switch must be configured in **trunk** port mode.
- FIP traffic uses the native VLAN.
- You can enable VN2VN_Port FIP snooping on a native VLAN.

ENode-Facing Interfaces

When the interfaces that belong to an FCoE VLAN connect directly to FCoE devices (there is no other transit switch between the FCoE devices and the switch), we recommend that you either enable VN2VN_Port FIP snooping on all FCoE VLANs to ensure secure connections between VN_Ports, or enable VN2VF_Port FIP snooping on FCoE VLANs that connect ENodes to an FC switch. FIP snooping should always be enabled at the access edge.
Systems that run Enhanced Layer 2 Software (ELS) support a slightly different configuration on ENode-facing interfaces than systems that do not run ELS. This section describes:

- Non-ELS Port Mode for FCoE Interfaces on page 49
- ELS Interface Mode for FCoE Interfaces on page 49
- Trusted and Untrusted FCoE Interfaces on page 49

### Non-ELS Port Mode for FCoE Interfaces

The interfaces that belong to FCoE VLANs (interfaces that connect to CNAs in FCoE devices) should be configured in **tagged-access** port mode, unless your CNA does not support tagged VN2VN traffic. After you enable VN2VN_Port FIP snooping on an FCoE VLAN, the transit switch denies FCoE traffic from any ENode on that VLAN until the ENode performs a valid fabric login (FIP FLOGI) with another ENode.

The **tagged-access** port mode was not available in Junos OS Release 11.3 and prior releases. In Release 11.3 and earlier, **trunk** port mode was used for Ethernet interfaces that connected to FCoE access devices. Because **tagged-access** mode is now available, using **trunk** mode for interfaces connected to FCoE CNAs is not recommended.

If an existing configuration uses **trunk** mode for ports connected to FCoE CNAs, you can change the port mode to **tagged-access** without disrupting traffic. Although we recommend changing the port mode from **trunk** to **tagged-access** as a best practice, it is not mandatory. New configurations should use **tagged-access** mode for interfaces that connect to FCoE devices.

### ELS Interface Mode for FCoE Interfaces

The interfaces that belong to FCoE VLANs (interfaces that connect to CNAs in FCoE devices) on systems that support ELS should be configured in **trunk** interface mode. After you enable VN2VF_Port FIP snooping on an FCoE VLAN, the transit switch denies FCoE traffic from any ENode on that VLAN until the ENode performs a valid fabric login with an FC switch.

### Trusted and Untrusted FCoE Interfaces

Do not configure ENode-facing interfaces as FCoE trusted interfaces when VN2VF_Port FIP snooping is enabled on those interfaces. If you enable VN2VF.Port FIP snooping on an FCoE VLAN and you configure ENode-facing interfaces that are members of the FIP snooping VLAN as **fcoe-trusted**, then FCoE devices might not be able to log in to the FC network.

Changing ports from untrusted to trusted removes any existing VN2VF.Port FIP snooping filters from the ports and terminates the existing sessions. Changing the fabric ports from trusted to untrusted forces all of the FCoE sessions on those ports to log out so that when the ENodes and VN_Ports log in again, the switch can build the appropriate VN2VF.Port FIP snooping filters.
Network-Facing Interfaces (Connecting to Another Transit Switch)

Configure any interface that is connected to another transit switch (not to an ENode) as an FCoE trusted interface, in trunk port mode, and as a 10-Gigabit Ethernet interface.

Network-facing Ethernet interfaces have the following requirements and behaviors:

- You must explicitly configure network-facing trunk ports on an FCoE transit switch as FCoE trusted interfaces.
- After you configure a network-facing trunk port as a trusted interface, the FCoE transit switch always processes frames from the connected switch because they come from a source on a trusted interface.
- As a best practice, configure ports in an FCoE VLAN as tagged access ports, but access and trunk port modes are also supported to accommodate whatever types of VN2VN traffic your CNA supports.

Beacon Period (VN2VN_Port FIP Snooping Link Maintenance)

The transit switch needs to maintain the virtual links between VN_Ports, and needs to know when sessions begin and end, and when to install and remove the FIP snooping filters. FIP snooping uses a FIP keepalive advertisement to accomplish this task. VN2VN_Port FIP snooping does not exchange FIP keepalive timer information. Instead, you configure a beacon period, which performs the same function as a keepalive timer.

The beacon period is the time interval between messages which verify that the connection is still valid and that the device at the other end of the virtual link is still reachable. You set the beacon period value for each FCoE VLAN that you configure to do VN2VN_Port FIP snooping.

NOTE: Explicitly set the beacon period when you configure VN2VN_Port FIP snooping. VN_Ports do not automatically send beacons.

ENodes transmit periodic multicast N_Port_ID beacons to the ALL-VN2VN-ENode-MACs address. The transmission period varies by a random delay of between 0 ms and 100 ms to avoid synchronized bursts of multicast traffic on the network.

If the transit switch does not receive a beacon message from an ENode within 2.5 times the configured beacon period, the transit switch considers the virtual link to be down and terminates the virtual link to that ENode.

QFabric System Differences in VN2VN_Port FIP Snooping Traffic Handling

Configuring VN2VN_Port FIP snooping on a QFabric system is the same as configuring VN2VN_Port FIP snooping on a standalone switch. However, there are internal differences in the way a QFabric system handles VN2VN_Port FIP snooping traffic compared to the way a standalone switch handles VN2VN_Port FIP snooping traffic. The internal differences are transparent. Whether you configure VN2VN_Port FIP snooping on a QFabric
system or on a standalone switch, the proper FIP snooping filters and forwarding information are installed on each device.

On standalone switches, the VN2VN_Port FIP snooping traffic does not cross a fabric (Interconnect device). VN2VN_Port traffic enters and exits ports on a single switch, so the ingress port and the egress port have access to the same local forwarding and FIP snooping databases.

However, on a QFabric system, VN2VN_Port FIP snooping traffic might enter on the ingress port of one Node device, traverse the Interconnect device fabric, and exit on the egress port of a different Node device. In this case, the QFabric system must ensure that the FIP snooping database and forwarding information for the VN2VN_Port traffic is installed correctly on both of the Node devices so that traffic is correctly filtered and forwarded.

For example, Figure 4 on page 51 shows that VN2VN_Port traffic from FCoE host ENode E1 enters the QFabric system at Node device ND1, traverses the Interconnect device fabric, and then exits from Node device ND2 before arriving at FCoE host ENode E2. Similarly, VN2VN_Port traffic from FCoE host ENode E2 enters the QFabric system at Node device ND2, traverses the Interconnect device fabric, and then exits from Node device ND1 before arriving at FCoE host ENode E1.

Figure 4: VN2VN_Port Traffic Across a QFabric Interconnect Device

When the QFabric system receives a FLOGI ACC from either ENode E1 or ENode E2, the QFabric system creates and installs the correct VN2VN_Port FIP snooping filters on both Node devices, and updates the forwarding tables accordingly.

In addition, the QFabric system must also ensure that the VN2VN_Port FIP snooping session statistics are correctly counted. Even though a session is running on each of the two Node devices, the QFabric system counts the complete VN2VN_Port connection as one session because the two Node devices belong to the same session. This ensures that VN2VN_Port sessions that traverse the Interconnect device fabric are counted as one unique session, not as two separate sessions.

Related Documentation
- Understanding DCB Features and Requirements on page 16
- Understanding FCoE Transit Switch Functionality on page 26
- Understanding VN_Port to VF_Port FIP Snooping on an FCoE Transit Switch on page 38
- Overview of FIP on page 9
- Understanding Fibre Channel Terminology on page 76
- Understanding FIP Snooping, FBF, and MVR Filter Scalability on page 53
- Configuring VLANs for FCoE Traffic on an FCoE Transit Switch on page 174
- Example: Configuring VN2VN_Port FIP Snooping (FCoE Hosts Directly Connected to the Same FCoE Transit Switch)
- Example: Configuring VN2VN_Port FIP Snooping (FCoE Hosts Directly Connected to Different FCoE Transit Switches)
- Example: Configuring VN2VN_Port FIP Snooping (FCoE Hosts Indirectly Connected Through an Aggregation Layer FCoE Transit Switch)
- Enabling VN2VN_Port FIP Snooping and Configuring the Beacon Period on an FCoE Transit Switch on page 182
Understanding FIP Snooping, FBF, and MVR Filter Scalability

The VLAN filter processor (VFP) ternary content addressable memory (TCAM) stores the VLAN filter configuration for three filter types:

- Fibre Channel over Ethernet (FCoE) Initialization Protocol (FIP) snooping—FIP snooping filters prevent an FCoE device from gaining unauthorized access to a Fibre Channel (FC) storage device or to another FCoE device. VN2VF_Port FIP snooping filters prevent an FCoE device from gaining unauthorized access to devices on an FC network. VN2VN_Port FIP snooping filters prevent an FCoE device from gaining unauthorized access to another FCoE device directly through the standalone switch or QFabric system, without traversing the FC network.

The VFP TCAM stores the VN2VF_Port and VN2VN_Port FIP snooping filters that the switch automatically creates when you enable FIP snooping on a VLAN that carries FCoE traffic. See “Understanding VN_Port to VF_Port FIP Snooping on an FCoE Transit Switch” on page 38 and “Understanding VN_Port to VN_Port FIP Snooping on an FCoE Transit Switch” on page 45 for more information.

- Filter-based forwarding (FBF)—FBF enables you to use firewall filters to direct packets to virtual routing instances. The switch then forwards the matching packets based on the configuration of the routing instances. The VFP TCAM stores the terms you configure for FBF filters. See Understanding Filter-Based Forwarding for more information.

- Multicast VLAN registration (MVR)—MVR enables you to configure a multicast source VLAN (MVLAN) that is shared across a Layer 2 network. An MVLAN distributes IPTV multicast streams across different VLANs without having to create a separate multicast stream for each VLAN, and without compromising the security and separation of traffic in the different VLANs. The VFP TCAM stores the MVR rules you configure for MVLANs. See Understanding Multicast VLAN Registration for more information.

FIP snooping filters, FBF filters, and MVR rules share the VFP TCAM memory space. In most use cases, the VFP TCAM memory is sufficient to store filter terms and information for all three applications.

VFP TCAM Architecture and Allocation

When packets arrive at an ingress interface, the VFP TCAM is the first TCAM in the packet pipeline. The VFP TCAM stores a total of 1024 entries. The 1024 entries are partitioned into four equal slices of 256 entries.

The VFP TCAM allocates entries to three filter types (FIP snooping filters, FBF filter terms, and MVR rules) in 256-entry slices. The VFP TCAM dynamically allocates the minimum number of memory slices required to store the filters for a particular filter type, as needed.
The TCAM does not allocate partial slices to a filter type, and slices cannot be shared among filter types. At any given time, each slice contains entries for one and only one filter type.

For example, if you configure one MVR rule, the system allocates a whole slice to MVR rules, even if the MVR rule consumes only one TCAM entry. The remaining 256 entries in the slice allocated to MVR rules can store subsequently configured MVR rules, but not FIP snooping or FBF filters. Similarly, if FIP snooping filters consume 50 entries of a 256-entry slice, the remaining 206 entries in the FIP snooping slice are available only to store more FIP snooping filters, not to store FBF filter terms or MVR rules.

The VFP TCAM allocates slices to a filter type only if there is at least one configured filter or rule for that filter type. If no filters exist for a filter type, then the VFP TCAM does not allocate a slice to that filter type.

NOTE: The VFP TCAM rejects partial filters. For example, if an FBF filter contains six terms, but there is only space in the TCAM for four of those terms, the whole filter is not committed.

Each filter type can use from zero slices to all four slices of VFP TCAM space. However, if one filter type uses three slices, then only one slice remains, so only one other filter type can use the remaining slice. In that situation, if you configure filters for all three filter types, the last filter type that you configure receives no TCAM space for its filter entries. Filters that receive no TCAM entry space are not implemented.

VFP TCAM Entry Consumption

FIP snooping filters, FBF filters, and MVR rules consume VFP TCAM entry space in different ways:

- FIP Snooping Filter VFP TCAM Consumption on page 54
- FBF Filter VFP TCAM Consumption on page 55
- MVR Filter VFP TCAM Consumption on page 56
- VFP TCAM Consumption Summary Table on page 56

FIP Snooping Filter VFP TCAM Consumption

VN2VF_Port FIP snooping filters consume VFP TCAM entry space differently than VN2VN_Port FIP snooping filters:

- VN2VF_Port FIP Snooping Filter VFP TCAM Consumption on page 55
- VN2VN_Port FIP Snooping Filter VFP TCAM Consumption on page 55

NOTE: One FCoE VLAN cannot support both VN2VF_Port traffic and VN2VN_Port traffic. Configure separate FCoE VLANs for VN2VF_Port traffic and for VN2VN_Port traffic.
**VN2VF Port FIP Snooping Filter VFP TCAM Consumption**

The switch uses an algorithm that allows one 256-entry slice of the VFP TCAM to store the maximum possible number of VN2VF Port FIP snooping filters (2500 filters). VN2VF Port FIP snooping filters never consume more than one slice of the VFP TCAM.

Regardless of whether there is one VN2VF Port FIP snooping session or there are 2500 VN2VF Port FIP snooping sessions, VN2VF Port FIP snooping filters consume one slice of the VFP TCAM. (If there are no VN2VF Port or VN2VN Port FIP snooping sessions, the TCAM does not allocate a slice for FIP snooping filters.)

**VN2VN Port FIP Snooping Filter VFP TCAM Consumption**

VN2VN Port FIP snooping filters consume one VFP TCAM entry for each VN2VN Port session. The maximum number of VN2VN Port FIP snooping sessions is 376 sessions per switch. (If you configure an interface that carries VN2VN Port FIP snooping traffic as a trusted interface, the switch does not apply filters on the trusted interface.)

Because the switch can have up to 376 VN2VN Port sessions running simultaneously, with each session consuming one entry, VN2VN Port FIP snooping filters consume VFP TCAM space as follows:

- 1–256 filters consume one slice
- 257–376 filters consume two slices

**FBF Filter VFP TCAM Consumption**

Each FBF filter term is double-wide, so each FBF filter term consumes two entries in the VFP TCAM. One 256-entry slice can contain up to 128 FBF filter terms. FBF filters consume VFP TCAM space as follows:

- 1–128 entries consume one slice
- 129–256 entries consume two slices
- 257–384 entries consume three slices
- 385–512 entries consume four slices

**NOTE:** In practice, FBF filters can consume only three slices of the VFP TCAM because FBF filters are also stored simultaneously in the ingress filter processor (IFP) TCAM, and the IFP TCAM can store only 384 FBF filter terms (768 entries, or 3 TCAM slices).

For example, if you configure FBF filters that contain 200 terms, then the FBF filters require 400 VFP TCAM entries and consume 2 slices.

FBF filter entries are simultaneously stored in the VFP TCAM and the IFP TCAM. The IFP TCAM can only contain up to 768 entries—256 fewer entries (1 slice) than the VFP TCAM. As with the VFP TCAM, FBF filters consume two IFP TCAM entries per filter term. In addition to FBF filter terms, the IFP TCAM stores filter entries for firewall filters.
CAUTION: There must be enough space in the VFP TCAM and the IFP TCAM for the FBF filter entries. If both TCAMs do not have enough space for the FBF filters, the switch rejects the portion of the configuration that it cannot store and sends a syslog message to notify you.

For example, if you configure FBF filters that have 400 terms, even though the VFP TCAM has enough space to store the resulting 800 entries, the switch rejects a portion of the configuration because the IFP TCAM can store a maximum of only 768 entries. If the IFP TCAM stores no other filter entries, the switch rejects 32 FBF filter entries.

In another example, if you configure firewall filters that have a total of 200 terms, which consume 200 entries in the IFP TCAM, and you then configure FBF filters that have a total of 300 terms, the switch rejects a portion of the configuration because the FBF filters require 600 entries. Combined with the 200 entries required for the firewall filters, the total number of 800 entries exceeds the maximum of 768 entries that the IFP TCAM can store. In this case, the switch accepts the first 768 entries and rejects the rest of the filter entries. The switch installs the filter entries in the order that they are committed; the rejected entries are the last entries the switch attempts to commit after the TCAM space is exhausted.

The IFP TCAM limit of 768 entries means that the true maximum number of FBF filter terms is 384 terms, even though the VFP TCAM can store up to 512 FBF terms.

MVR Filter VFP TCAM Consumption

Each MVR rule consumes one entry in the VFP TCAM, so MVR rules consume VFP TCAM space as follows:

- 1–256 rules consume one slice
- 257–512 rules consume two slices
- 513–758 rules consume three slices
- 759–1024 rules consume four slices

VFP TCAM Consumption Summary Table

Table 4 on page 57 summarizes VFP TCAM consumption.

NOTE: FBF filters are simultaneously stored in the VFP TCAM and in the IFP TCAM. Due to the IFP TCAM limit of 768 entries (384 FBF filters), which is 256 entries fewer than the VFP TCAM, the effective VFP TCAM consumption limit for FBF filters is lower than the total amount of VFP TCAM entry space, even when no other filters consume VFP TCAM space.
Table 4: VFP TCAM Entry Consumption Summary

<table>
<thead>
<tr>
<th>Filter Type</th>
<th>VFP TCAM Entry Consumption</th>
<th>Maximum VFP TCAM Slices Consumed</th>
<th>Other Limitations</th>
</tr>
</thead>
<tbody>
<tr>
<td>VN2VF_Port FIP snooping</td>
<td>Never consumes more than one slice</td>
<td>One slice (regardless of number</td>
<td>2500 session maximum</td>
</tr>
<tr>
<td>filters</td>
<td></td>
<td>of sessions)</td>
<td></td>
</tr>
<tr>
<td>VN2VN_Port FIP snooping</td>
<td>One entry per session</td>
<td>Two</td>
<td>376 session maximum</td>
</tr>
<tr>
<td>filters</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FBF filters</td>
<td>Two entries per filter</td>
<td>Three (due to IFP TCAM limitation)</td>
<td>384 filters (due to IFP TCAM limitation)</td>
</tr>
<tr>
<td>MVR rules</td>
<td>One entry per rule</td>
<td>Four</td>
<td>1024 rule maximum</td>
</tr>
</tbody>
</table>

Rejected Filter Configurations (No Available VFP TCAM Space)

If there is not enough space available in the VFP TCAM to store the FIP snooping filters, the configured FBF filters, and the MVR rules, the switch rejects only the portion of the configuration that it cannot store. Any portion of the filter configuration that the TCAM can store, is stored. In most cases, even if the switch rejects part of the configuration, part of the configuration is also stored.

If the switch rejects any portion of a configuration, the switch sends a syslog message to notify you of the failure. The switch does not generate a commit error, and the rejected portion of the configuration remains on the switch, even though the rejected configuration does not function. (The accepted portions of the configuration function as expected.) The syslog message shows you the filter configuration that the switch rejected.

We strongly recommend that you always delete rejected filter configurations from the switch. It is important to delete rejected filter configurations because:

- Even though the rejected configuration remains on the switch, it does not function.
- After a reboot, there is no guarantee that the same filters will be rejected. The previously rejected filters might be accepted, and other filters that had previously been accepted might be rejected. Therefore, the functioning filter configuration could be changed inadvertently and unexpectedly.
- Even if a VFP TCAM slice becomes available, the switch does not automatically allocate the available slice to the rejected configuration. To use the available slice, you must delete and reconfigure the rejected configuration.

For example, you configure FBF filters and MVR rules on a switch, and that switch also transports FCoE traffic with VN2VF_Port FIP snooping (never consumes more than one slice) enabled on FCoE access interfaces. After you commit the configuration, you check the syslog. You find that the VN2VF_Port FIP snooping and FBF filters consume all four slices of the VFP TCAM, and the MVR configuration was rejected. Instead of deleting the MVR configuration, you leave it on the switch. Subsequently, all VN2VF_Port FIP snooping sessions end, the FIP snooping filters time out and are removed from the VFP TCAM, so the slice that was allocated to VN2VF_Port FIP snooping filters becomes free. However, the MVR rules do not automatically receive the free slice.
To force the switch to allocate the free slice to the MVR rules, you should delete the MVR rules from the configuration and then reconfigure the MVR rules. When you commit the new configuration, check the syslog messages to ensure that the MVR rule configuration was accepted.

In this example, you could also choose to free a VFP TCAM slice for MVR rule storage by deleting some of the FBF filters. To do this, you delete both the unneeded FBF filters and the MVR rule configuration. Then you reconfigure the MVR rules, and check the syslog to ensure that the configuration was successful.

### VFP TCAM Allocation and Consumption (Scaling) Examples

The following examples illustrate how FIP snooping entries, FBF filter entries, and MVR rule entries consume VFP TCAM slices:

- Example 1: Three Filter Types Consume Three Slices on page 58
- Example 2: Three Filter Types Consume Four Slices on page 58
- Example 3: Two Filter Types Consume Four Slices on page 59
- Example 4: Three Filter Types Oversubscribe the VFP TCAM on page 59

#### Example 1: Three Filter Types Consume Three Slices

Filters and rules are configured in the following sequence:

- 100 VN2VN_Port FIP snooping filters (1 slice)
- 2 MVR rules (1 slice, 2 entries)
- 60 FBF filter terms (1 slice, 120 entries)

One slice remains free. The slice allocated to VN2VN_Port FIP snooping filters can store 156 more filters before another slice is required. The slice allocated to MVR rules can store 254 more rules before another slice is required. The slice allocated to FBF filters can store 68 more filter terms (136 entries) before another slice is required. Providing that the IFP TCAM has space for the FBF filter terms, the switch accepts this configuration and rejects no filters.

#### Example 2: Three Filter Types Consume Four Slices

Filters and rules are configured in the following sequence:

- 2000 VN2VF_Port FIP snooping filters (always 1 slice)
- 18 MVR rules (1 slice, 18 entries)
- 150 FBF filter terms (2 slices, 300 entries)

All four slices are allocated to filter types. The slice allocated to MVR rules can store 238 more rules before it is full. The slice allocated to FBF filters can store 106 more filter terms (212 entries) before it is full. Providing that the IFP TCAM has space for the FBF filter terms, the switch accepts this configuration and rejects no filters.
NOTE: If you configure more MVR rules or FBF filters than entry space remaining in the slices, the switch rejects those rules and filters because no slice is available. The switch installs filters in the order that they were configured, so if filters are rejected, the filters configured last are rejected.

Example 3: Two Filter Types Consume Four Slices
Filters and rules are configured in the following sequence:

- 50 VN2VF_Port FIP snooping filters (always 1 slice)
- 300 FBF filter terms (3 slices, 600 entries)

All four slices are allocated to filter types. No slices are available for MVR rules. The third slice allocated to FBF filters can store 84 more filter terms (168 entries) before it consumes all of its entry space. Providing that the IFP TCAM has space for the FBF filter terms, the switch accepts this configuration and rejects no filters.

NOTE: If you configure MVR rules or if you configure more than 84 more FBF filters, the switch rejects those rules and filters because no slice is available for the MVR rules, and the FBF filter slice has entry space for only 84 more filter terms.

Example 4: Three Filter Types Oversubscribe the VFP TCAM
Filters and rules are configured in the following sequence:

- 1750 VN2VF_Port FIP snooping filters (always 1 slice)
- 10 MVR rules (1 slice, 10 entries)
- 275 FBF filter terms (2 slices, 512 accepted entries, 38 rejected entries)

All four slices are allocated to filter types. The slice allocated to MVR rules can store 246 more rules before it is full, but the number of FBF filter terms exceeds the amount of available VFP TCAM storage space. (The 275 FBF filter terms consume 550 VFP TCAM entries. However, there are only two available slices, for a total of 512 available entry spaces, so only 256 FBF filter terms can be stored, leaving 19 rejected FBF filter terms.)

The switch accepts the VN2VF_Port FIP snooping filters, the MVR rules, and 256 FBF filter terms. The switch retains the excess FBF filters in the configuration, but does not install those filters in the VFP TCAM. In this case, you delete the rejected FBF filter terms from the configuration. Alternatively, you could delete the MVR rules from the configuration to free a slice of the TCAM, and then delete and reconfigure the rejected FBF filters so that the system allocates the freed slice to the FBF filters.
NOTE: The sequence of configuration makes a difference; if there is not enough VFP TCAM space for a given filter type, the switch installs the filters that fit in the order they are configured. For example, if you configure the FBF filters before you configure the MVR rules, the VFP TCAM allocates one slice to FIP snooping filters, three slices to FBF filters (assuming the IFP TCAM has available space), and no slices to MVR rules, because all four slices are allocated before the switch attempts to install the MVR rules in the VFP TCAM.

Filter Configuration Recommendations

To utilize the VFP TCAM space most efficiently:

- Configure and Maintain the Fewest Number of Filters Needed on page 60
- Always Delete Rejected Filter Configurations on page 61

Configure and Maintain the Fewest Number of Filters Needed

To conserve VFP TCAM entry space, and because FBF filter storage also depends on the availability of IFP TCAM space, we recommend that you configure as few FBF filters and MVR rules as is practical to serve your network needs. The more filters you configure, the greater the possibility of exceeding TCAM storage capacity.

Several factors determine VFP TCAM consumption:

- Type of filters configured—Different filter types consume different amounts of VFP TCAM space. VN2VF_Port FIP snooping filters never consume more than one slice. MVR rules and VN2VN_Port FIP snooping filters consume entries in a slice at a rate of one entry per MVR rule or VN2VN_Port session. FBF filter terms consume entries in a slice at a rate of two entries per FBF filter term.

- Number of filters configured—Although the number of filters does not affect the number of slices allocated to the VN2VF_Port FIP snooping filter type (it is always one slice for one or more VN2VF_Port FIP snooping filters and no slice for no FIP snooping filters), the number of VN2VN_Port FIP snooping filters, MVR rules, and FBF filter terms that you configure determine how many VFP TCAM slices are required for each filter type. For example, if you configure 257 MVR rules, the MVR rule entries consume 2 slices. One slice stores 256 MVR rules (entries), and one slice stores 1 MVR rule (entry). In this case, if you can eliminate one MVR rule, you can free a slice to allocate to other filter types.

- Sequence of filter configuration—if you configure too many filters for the VFP TCAM to store, the last filters you configure are not stored in the TCAM.

Always check the syslog after you configure FBF filters or MVR rules to ensure that the configuration was not rejected. If you enable FIP snooping on access ports, check the syslog to ensure that the configuration was not rejected due to lack of VFP TCAM space.
If you check the syslog and a filter configuration has been rejected, delete the filters that were rejected from the configuration.

**TIP:** If you no longer need an FBF filter or an MVR rule, delete it from the configuration to conserve VFP TCAM space. Enable VN2VF_Port or VN2VN_Port FIP snooping on access ports only if the switch port is directly connected to FCoE devices. (FIP snooping should be performed at the access edge. FIP snooping should not be performed on traffic that has already been snooped and filtered at the access edge. If another switch that is physically between the transit switch (or QFabric system) and the FCoE devices already performs FIP snooping, you do not have to enable FIP snooping on the transit switch or QFabric system, but you can.)

**Always Delete Rejected Filter Configurations**

The switch does not return a commit error if it rejects any portion of a configuration. Instead, the switch sends a syslog message to report the rejected portion of the configuration. The rejected portion of the configuration remains on the switch, but does not function.

After you configure FBF filters or MVR rules, or enable FIP snooping, check the syslog messages to ensure that the switch accepted the configuration. If the switch rejected any portion of the configuration, delete that portion of the configuration. (You do not need to delete the portion of the configuration that was accepted, unless you want to reconfigure those filters or rules.)

**CAUTION:** If you do not delete rejected filter configurations, and if you reboot the system, you cannot predict which filters the system installs after the reboot. For example, a switch with the following configuration has more configured filters than the VFP TCAM can support:

- VN2VF_Port FIP snooping sessions (always consumes one slice)
- 20 MVR rules (consume one slice)
- 300 FBF filters (attempt to consume three slices, but because only two slices are available, 256 filters consume two slices, and the remaining 44 filters are rejected)

If you do not delete the 44 rejected FBF filters, then if the switch reboots, the 44 FBF filters that were rejected might be accepted, and 44 different FBF filters might be rejected. This unpredictable behavior is the reason that you should check the syslog messages after you configure filters, and if any filters were rejected, you should always delete the rejected filters from the configuration.

**Related Documentation**  
- Understanding VN_Port to VF_Port FIP Snooping on an FCoE Transit Switch on page 38
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 passthrough transit switch ports.

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

This topic describes:

- Supported Topology on page 63
- FIP Snooping and FCoE Trusted Ports on page 64
- CoS and Data Center Bridging (DCB) on page 65
Supported Topology

Switches that are not directly connected to FCoE hosts and that act as passthrough transit switches support MC-LAGs for FCoE traffic in an inverted-U network topology. Figure 5 on page 63 shows an inverted-U topology using QFX3500 switches.

The following rules and guidelines apply to MC-LAGs when used for FCoE traffic. The rules and guidelines help 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 passthrough 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 not 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.)

- 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. In essence, 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. In essence, 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 that 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 5 on page 63, 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 5 on page 63, 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.

LLDP and 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.

Related Documentation
- Understanding Multichassis Link Aggregation
- Example: Configuring CoS for FCoE Transit Switch Traffic Across an MC-LAG
- Example: Configuring Multichassis Link Aggregation
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
- IEEE 802.1Qbb priority-based flow control (PFC)

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

NOTE: For end-to-end congestion control, 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 is mapped to a 3-bit IEEE 802.1p CoS code point flag 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).

Video: Why Use PFC in a Data Center Network?

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 specified types of traffic (for example, Fibre Channel over Ethernet 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.

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

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

- Ethernet PAUSE on page 67
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. 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 interface-name ether-options` 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 interface-name ether-options` 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, the PFC configuration overrides Ethernet PAUSE flow control.) Both symmetric and asymmetric flow control are supported.

- Symmetric Flow Control on page 68
- Asymmetric Flow Control on page 68
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 5 on page 68 describes the configured flow control state when you set the receive (Rx) and transmit (Tx) buffers on an interface:

<table>
<thead>
<tr>
<th>Receive (Rx) Buffer</th>
<th>Transmit (Tx) Buffer</th>
<th>Configured Flow Control State</th>
</tr>
</thead>
<tbody>
<tr>
<td>On</td>
<td>Off</td>
<td>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).</td>
</tr>
<tr>
<td>Off</td>
<td>On</td>
<td>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.)</td>
</tr>
<tr>
<td>On</td>
<td>On</td>
<td>Same functionality as symmetric Ethernet PAUSE. Interface generates and sends Ethernet PAUSE messages and responds to received Ethernet PAUSE messages.</td>
</tr>
<tr>
<td>Off</td>
<td>Off</td>
<td>Ethernet PAUSE flow control is disabled.</td>
</tr>
</tbody>
</table>
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 6 on page 69:

Table 6: Flow Control State Advertised to the Connected Peer (Autonegotiation)

<table>
<thead>
<tr>
<th>Rx Buffer State</th>
<th>Tx Buffer State</th>
<th>PAUSE Bit</th>
<th>ASM_DIR Bit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Off</td>
<td>Off</td>
<td>0</td>
<td>0</td>
<td>The interface advertises no Ethernet PAUSE capability. This is equivalent to disabling flow control on an interface.</td>
</tr>
<tr>
<td>On</td>
<td>On</td>
<td>1</td>
<td>0</td>
<td>The interface advertises symmetric flow control (both the transmission of Ethernet PAUSE messages and the ability to receive and respond to Ethernet PAUSE messages).</td>
</tr>
<tr>
<td>On</td>
<td>Off</td>
<td>0</td>
<td>1</td>
<td>The interface advertises asymmetric flow control (the transmission of Ethernet PAUSE messages, but not the ability to receive and respond to Ethernet PAUSE messages).</td>
</tr>
<tr>
<td>Off</td>
<td>On</td>
<td>1</td>
<td>1</td>
<td>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.)</td>
</tr>
</tbody>
</table>

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 7 on page 70. 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 7: Asymmetric Ethernet PAUSE Behavior on Local and Peer Interfaces

<table>
<thead>
<tr>
<th>Local Interface (QFX Series or EX4600 Switch)</th>
<th>Peer Interface</th>
<th>Local Resolution</th>
<th>Peer Resolution</th>
</tr>
</thead>
<tbody>
<tr>
<td>PAUSE Bit</td>
<td>ASM_DIR Bit</td>
<td>PAUSE Bit</td>
<td>ASM_DIR Bit</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>Don't care</td>
<td>Don't care</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>Don't care</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>Don't care</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>Don't care</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>Don't care</td>
<td>Don't care</td>
</tr>
</tbody>
</table>

**NOTE:** For your convenience, Table 7 on page 70 replicates Table 28B-3 of Section 2 of the IEEE 802.3 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 with 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 Fibre Channel over Ethernet (FCoE), LAN backup, or management, while using standard frame-drop congestion management for IP traffic on the same link.

Potential consequences of link-level 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 8 on page 72:

Table 8: Default PFC Priority to Queue and Forwarding Class Mapping

<table>
<thead>
<tr>
<th>IEEE 802.1p Priority (Code Point)</th>
<th>Queue</th>
<th>Forwarding Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 (000)</td>
<td>0</td>
<td>best-effort</td>
</tr>
<tr>
<td>1 (001)</td>
<td>1</td>
<td>best-effort</td>
</tr>
<tr>
<td>2 (010)</td>
<td>2</td>
<td>best-effort</td>
</tr>
<tr>
<td>3 (011)</td>
<td>3</td>
<td>fcoe</td>
</tr>
<tr>
<td>4 (100)</td>
<td>4</td>
<td>no-loss</td>
</tr>
<tr>
<td>5 (101)</td>
<td>5</td>
<td>best-effort</td>
</tr>
<tr>
<td>6 (110)</td>
<td>6</td>
<td>network-control</td>
</tr>
<tr>
<td>7 (111)</td>
<td>7</td>
<td>network-control</td>
</tr>
</tbody>
</table>

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 forwarding class 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.

You enable PFC on a priority by:
1. Specifying the IEEE 802.1p code point to pause in the input stanza of a CNP
2. Applying 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 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.

Although unicast traffic and multidestination (multicast, broadcast, and destination lookup fail) traffic must use different classifiers, you can map a unicast queue (queue 0 through 7) and a multidestination queue (queue 8, 9, 10, or 11) to the same PFC priority so that both unicast and multicast traffic use that priority. Do not map multidestination traffic to lossless priorities. 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: 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
- Overview of CoS Changes Introduced in Junos OS Release 12.2
- Understanding CoS IEEE 802.1p Priorities for Lossless Traffic Flows
- Understanding CoS IEEE 802.1p Priority Remapping on an FCoE-FC Gateway
- Understanding DCB Features and Requirements on page 16
Understanding Fibre Channel Terminology

To understand the Fibre Channel (FC) and Fibre Channel over Ethernet (FCoE) capabilities of the switches, you should become familiar with the terms defined in Table 9 on page 76.

Table 9: Fibre Channel Terms

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>addressing mode</td>
<td>Format for the locally unique MAC address the FC switch assigns to FCoE devices for FCoE transactions after FIP establishes a connection between an FCoE device and the FC switch. The two addressing modes are fabric-provided MAC address (FPMA) and server-provided MAC address (SPMA). Only FPMA is supported. During FLOGI or FDISC, the ENode advertises the addressing modes it supports. If the FC switch supports an addressing mode that the ENode uses, the virtual link can be established, and the devices can communicate. See also fabric-provided MAC address (FPMA) and server-provided MAC address (SPMA).</td>
</tr>
<tr>
<td>ALL-ENode-MACs</td>
<td>Well-known multicast MAC address to which all FCoE ENodes listen. FCFs send multicast FIP discovery advertisement messages and FIP keepalive messages to the ALL-ENode-MACs address so that ENodes can discover and maintain connections to FCFs. The hexadecimal format of the address is 01:10:18:01:00:01. See also well-known address (WKA).</td>
</tr>
<tr>
<td>ALL-FCF-MACs</td>
<td>Well-known multicast MAC address to which all FCFs listen. ENodes send multicast FIP discovery solicitation messages to the ALL-FCF-MACs address to find out which FCFs can accept a login. The hexadecimal format of the address is 01:10:18:01:00:02. See also well-known address (WKA).</td>
</tr>
<tr>
<td>congestion notification</td>
<td>See quantized congestion notification (QCN).</td>
</tr>
<tr>
<td>converged network adapter (CNA)</td>
<td>Physical adapter that combines the functions of a Fibre Channel host bus adapter (HBA) to process FCoE frames and a lossless Ethernet network interface card (NIC) to process non-FCoE Ethernet frames. CNAs have one or more Ethernet ports. CNAs encapsulate Fibre Channel frames in Ethernet for FCoE transport and de-encapsulate Fibre Channel frames from FCoE to native Fibre Channel. See also host bus adapter (HBA).</td>
</tr>
</tbody>
</table>
Table 9: Fibre Channel Terms (continued)

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>data center bridging (DCB)</td>
<td>Set of IEEE specifications that enhance Ethernet to allow it to support converged Ethernet (LAN) and Fibre Channel (SAN) traffic on one Ethernet network. DCB features include priority-based flow control (PFC), enhanced transmission selection (ETS), Data Center Bridging Capability Exchange protocol (DCBX), quantized congestion notification (QCN), and full-duplex 10-Gigabit Ethernet ports. See also priority-based flow control (PFC), Ethernet PAUSE, enhanced transmission selection (ETS), Data Center Bridging Capability Exchange protocol (DCBX), and quantized congestion notification (QCN).</td>
</tr>
<tr>
<td>expansion port (E_Port)</td>
<td>An expansion port in an FC switch/FCF that connects the FC switch/FCF to the E_Port of another FC switch/FCF to form an Interswitch Link (ISL) in a common FC fabric.</td>
</tr>
<tr>
<td>Data Center Bridging Capability Exchange protocol (DCBX)</td>
<td>Discovery and exchange protocol for conveying configuration and capabilities among neighbors to ensure consistent configuration across the network. It is an extension of the Link Layer Data Protocol (LLDP, described in IEEE 802.1AB) See also data center bridging (DCB).</td>
</tr>
<tr>
<td>enhanced transmission selection (ETS)</td>
<td>Mechanism that provides finer granularity of bandwidth management within a link. See also data center bridging (DCB).</td>
</tr>
<tr>
<td>ENode</td>
<td>See FCoE Node (ENode)</td>
</tr>
<tr>
<td>ENode MAC</td>
<td>Lossless Ethernet MAC paired with an FCoE controller in an ENode. See also FCoE node (ENode).</td>
</tr>
<tr>
<td>ENode MAC address</td>
<td>Globally unique address assigned to the CNA by the manufacturer and used to identify the node for FIP transactions.</td>
</tr>
<tr>
<td>Ethernet PAUSE</td>
<td>As defined in IEEE 802.3X, a flow control mechanism that temporarily stops the transmission of Ethernet frames on a link for a specified period. A receiving element sends an Ethernet PAUSE frame when a sender transmits data faster than the receiver can accept it. Ethernet PAUSE affects the entire link, not just an individual flow. An Ethernet PAUSE frame temporarily stops all traffic transmission on the link and allows the receiver’s input buffer to empty sufficiently to restart traffic on the link. Ethernet PAUSE messages are sent to the previous hop and do not automatically propagate to the source of the congestion. See also priority-based flow control (PFC).</td>
</tr>
<tr>
<td>fabric</td>
<td>Interconnection of network nodes using one or more network switches that function as a network single logical entity.</td>
</tr>
<tr>
<td>Term</td>
<td>Definition</td>
</tr>
<tr>
<td>------------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>fabric discovery (FDISC)</td>
<td>Subsequent logins from the same ENode for different users, applications, or virtual machines after an ENode performs an initial FLOGI to log in to a switch. FC and FIP FDISC messages serve the same function in FC and FCoE networks, respectively. N_Ports send FC FDISC messages to the FC switch and VN_Ports send FIP FDISC messages to the FCF. After an N_Port acquires its initial N_Port ID through the FC FLOGI process, it can acquire additional N_Port IDs by sending an FC FDISC with a new worldwide port name and a source ID of 0x00000000. The new port name and blank source ID tell the FC switch to assign a new N_Port ID to the N_Port. The different N_Port IDs allow multiple virtual machines or users on the N_Port to have separate, secure virtual links on the same physical N_Port. These additional ports are also referred to as VN_Ports. FIP FDISC works the same way, except the VN_Port logs in using a FIP FLOGI message. See also fabric login (FLOGI) and N_Port ID.</td>
</tr>
<tr>
<td>fabric login (FLOGI)</td>
<td>Creation of a logical connection to the FC switch and establishment of a node’s operating environment. For FC devices, an N_Port logs in to the FC network by sending an FC FLOGI message to the F_PORT of an FC switch. For FCoE devices, a VN_Port logs in to the FC network by sending a FIP FLOGI message to the VF_PORT of an FC switch.</td>
</tr>
<tr>
<td>fabric port (F_PORT)</td>
<td>FC port on an FC switch or an FCF that connects point-to-point to an FC node port (N_Port) on an FC host (server or storage device). An F_PORT provides access to fabric services for FC devices. F_Ports are intermediate ports in a connection between FC device end-point N_Ports. For example, a connection between an FC host server and an FC storage device through an FC switch looks like this: FC server N_Port to FC switch ingress F_PORT to FC switch egress F_PORT to FC storage device N_Port. See also node port (N.Port).</td>
</tr>
<tr>
<td>fabric-provided MAC address (FPMA)</td>
<td>MAC address that an FCF assigns to a single ENode MAC through the FLOGI or FDISC process that is unique to the local fabric. The FPMA uniquely identifies a single VN_Port at that ENode MAC in FCoE transactions with the FCF. Because an ENode can have more than one ENode MAC, an FCF can assign multiple FPMAs to an ENode, one FPMA per ENode MAC. An FPMA is a 48-bit value that consists of two 24-bit values, the N_Port ID and the FC-MAP value. The N_Port ID uniquely identifies the VN_Port and the FC-MAP value identifies the FCF. See also FCoE node (ENode), N_Port ID, and FCoE mapped address prefix (FC-MAP).</td>
</tr>
<tr>
<td>FCF-MAC</td>
<td>Lossless Ethernet MAC paired with an FCoE controller in an FCF. The FCF-MAC enables the FCF to handle FCoE traffic.</td>
</tr>
</tbody>
</table>
## Table 9: Fibre Channel Terms (continued)

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FCoE controller</strong></td>
<td>Instantiates and terminates VN_Port and VF_Port instances on an ENode. An</td>
</tr>
<tr>
<td></td>
<td>ENode can have more than one FCoE controller. Each FCoE controller is paired</td>
</tr>
<tr>
<td></td>
<td>with a lossless Ethernet MAC on the ENode.</td>
</tr>
<tr>
<td></td>
<td>See also <em>lossless Ethernet MAC</em>.</td>
</tr>
<tr>
<td><strong>FC forwarder (FCF)</strong></td>
<td>Alternative term and acronym to refer to an FC switch that has all physical</td>
</tr>
<tr>
<td></td>
<td>Fibre Channel ports and the necessary set of services as defined in the T11</td>
</tr>
<tr>
<td></td>
<td>Organization Fibre Channel Switched Fabric (FC-SW) standards.</td>
</tr>
<tr>
<td><strong>FCoE forwarder (FCF)</strong></td>
<td>Defined by the Fibre Channel Backbone - 5 (FC-BB-5) Rev 2.00 specification</td>
</tr>
<tr>
<td></td>
<td>device that has the necessary set of services as defined in FC-SW and the</td>
</tr>
<tr>
<td></td>
<td>FCoE capabilities to act as an FCoE-based FC switch.</td>
</tr>
<tr>
<td><strong>FCoE Initialization Protocol (FIP)</strong></td>
<td>Layer 2 protocol for endpoint discovery, fabric login, and fabric association. FIP enables FCoE devices and FC switches to discover one another. Through FIP, FCoE nodes can log in to an FC switch, access the SAN FC fabric, and communicate with target FC devices. FIP messages also maintain the connection between the FCoE initiator and the FCF. FIP has its own EtherType (0x8914) to distinguish its traffic from payload-carrying FCoE traffic and other Ethernet traffic.</td>
</tr>
<tr>
<td><strong>FCoE link endpoint (LEP)</strong></td>
<td>Virtual FC interface mapped onto a physical Ethernet interface to handle FC</td>
</tr>
<tr>
<td></td>
<td>frame encapsulation and de-encapsulation and transmission and reception of</td>
</tr>
<tr>
<td></td>
<td>FC frames encapsulated in Ethernet through a single virtual link.</td>
</tr>
<tr>
<td><strong>FCoE mapped address prefix (FC-MAP)</strong></td>
<td>24-bit value that identifies the FC switch and is half of the 48-bit FPMA MAC address. The FC-MAP value can be configured on the FC switch and has a default value of 0EFC00h. The FC-MAP value was originally called the Fibre Channel Organizationaly Unique Identifier (FC-OUI). See also <em>fabric-provided MAC address (FPMA)</em>.</td>
</tr>
<tr>
<td><strong>FCoE node (ENode)</strong></td>
<td>Fibre Channel node that has one or more lossless Ethernet MACs, each paired</td>
</tr>
<tr>
<td></td>
<td>with an FCoE Controller in order to transmit FCoE frames. An ENode combines</td>
</tr>
<tr>
<td></td>
<td>FCoE termination functions and the FC stack on a CNA. ENodes present virtual</td>
</tr>
<tr>
<td></td>
<td>FC interfaces to FC switches or FCFs in the form of VN_Ports, which can</td>
</tr>
<tr>
<td></td>
<td>establish FCoE virtual links with FC switch/FCF VF_Ports. ENodes perform</td>
</tr>
<tr>
<td></td>
<td>FCoE related functions in a converged network adapter (CNA).</td>
</tr>
<tr>
<td></td>
<td>See also <em>converged network adapter (CNA)</em>.</td>
</tr>
<tr>
<td><strong>FCoE-FC gateway</strong></td>
<td>A form of N_Port virtualizer in which the node-facing ports are FCoE ports</td>
</tr>
<tr>
<td></td>
<td>and the FC switch-facing ports are FC ports.</td>
</tr>
<tr>
<td><strong>FCoE-FCoE gateway</strong></td>
<td>A form of N_Port virtualizer in which the node-facing ports are FCoE ports</td>
</tr>
<tr>
<td></td>
<td>and the FC switch-facing ports are FCoE ports.</td>
</tr>
<tr>
<td><strong>FC-FC gateway</strong></td>
<td>A form of N_Port virtualizer in which the node-facing ports are FC ports</td>
</tr>
<tr>
<td></td>
<td>and the FC switch-facing ports are FC ports.</td>
</tr>
</tbody>
</table>
Table 9: Fibre Channel Terms (continued)

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>FCoE transit switch (also known as a FIP snooping bridge)</td>
<td>Switch with a minimum set of features designed to support FCoE Layer 2 forwarding and FCoE security. The switch can also have optional additional features. Minimum feature support is:</td>
</tr>
<tr>
<td></td>
<td>• Priority-based flow control (PFC)</td>
</tr>
<tr>
<td></td>
<td>• Enhanced transmission selection (ETS)</td>
</tr>
<tr>
<td></td>
<td>• Data Center Bridging Capability Exchange Protocol (DCBX), including the FCoE application TLV</td>
</tr>
<tr>
<td></td>
<td>• FIP snooping (minimum support is FIP automated filter programming at the ENode edge)</td>
</tr>
<tr>
<td></td>
<td>Additional FIP snooping capabilities can include learning the virtual FC connection paths (VN2VF, VN2VN, or VE2VE) and monitoring the FIP keepalive mechanisms. Other optional capabilities can also enhance FCoE within the standards. FIP snooping is typically configurable on a per-VLAN basis.</td>
</tr>
<tr>
<td></td>
<td>A transit switch has an FC stack even though it is not an FC switch or an FCF.</td>
</tr>
<tr>
<td>FCoE VLAN</td>
<td>VLAN dedicated to carrying only FCoE traffic. FCoE traffic must travel in a VLAN. Only FCoE interfaces should be members of an FCoE VLAN. Ethernet traffic that is not FCoE traffic must travel in a different VLAN.</td>
</tr>
<tr>
<td>Fibre Channel</td>
<td>High-speed network technology used for storage area networks (SANs).</td>
</tr>
<tr>
<td>Fibre Channel fabric</td>
<td>Network of Fibre Channel devices that allows communication among devices, device name lookup, security, and redundancy.</td>
</tr>
<tr>
<td></td>
<td>Also a local fabric on a QFX3500 switch with FCoE interfaces connected to FCoE devices on the Ethernet network and native FC interfaces connected to an FC switch in a SAN.</td>
</tr>
<tr>
<td>Fibre Channel ID (FCID)</td>
<td>24-bit value the FC switch assigns to the N_Port or VN_Port as a unique identifier within the local FC network. The FCID consists of an 8-bit domain value, an 8-bit area value, and an 8-bit port value. The FCID is sometimes called an N_Port ID. See also N_Port ID.</td>
</tr>
<tr>
<td>Fibre Channel over Ethernet (FCoE)</td>
<td>Standard for transporting FC frames over Ethernet networks. FCoE encapsulates Fibre Channel frames in Ethernet so that the same high-speed Ethernet physical infrastructure can transport both data and storage traffic while preserving the lossless CoS that FC requires. FCoE has its own EtherType (0x8906) to differentiate it from other Ethernet traffic. FCoE runs on a DCB network. FCoE servers connect to a switch that supports both FCoE and native FC protocols. This allows FCoE servers on the Ethernet network to access FC storage devices in the SAN fabric on one converged network. See also data center bridging (DCB).</td>
</tr>
</tbody>
</table>
### Table 9: Fibre Channel Terms (continued)

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Fibre Channel services</strong></td>
<td>Functions required for establishing FC network connectivity among devices and for managing devices on the FC network, such as login servers, domain managers, name servers, and zone servers.</td>
</tr>
<tr>
<td><strong>FC stack</strong></td>
<td>FC or FCoE protocol capability implemented on a device to support the FC or FCoE functionality. Having an FC stack does not imply consuming a domain ID.</td>
</tr>
<tr>
<td></td>
<td>Each FC or FCoE enabled server or storage device has an FC stack. Similarly, an FC or FCoE switch, an FCF, an FCoE-FC gateway, and an FCoE transit switch have FC stacks.</td>
</tr>
<tr>
<td><strong>Fibre Channel switch</strong></td>
<td>Network switch that implements the Fibre Channel protocol.</td>
</tr>
<tr>
<td><strong>FIP discovery advertisement</strong></td>
<td>Multicast or unicast message that the FC switch (or FCF) transmits to ENodes to advertise the switch’s presence on the network so that ENodes can discover the switch and request to log in to the FC fabric.</td>
</tr>
<tr>
<td></td>
<td>The FC switch periodically sends multicast FIP discovery advertisements to the ALL-ENode-MACs address, a well-known address to which all ENodes listen. The multicast messages advertise the FC switch to all ENodes on the VLAN and serve as keepalive messages to maintain connectivity between the FC switch and ENodes.</td>
</tr>
<tr>
<td></td>
<td>When an ENode sends a FIP discovery solicitation message to the FC switch, the FC switch responds with a unicast FIP discovery advertisement to that ENode.</td>
</tr>
<tr>
<td><strong>FIP discovery solicitation</strong></td>
<td>Multicast or unicast message that an ENode transmits to FC switches (or FCFs) to find compatible switches in the network.</td>
</tr>
<tr>
<td></td>
<td>When an ENode initializes, it sends a multicast FIP discovery solicitation to the ALL-FCF-MACs address, a well-known address to which all FC switches and FCFs listen. Compatible switches reply with a unicast FIP discovery advertisement.</td>
</tr>
<tr>
<td></td>
<td>The ENode compiles a list of compatible switches, selects a switch, and logs in to that switch.</td>
</tr>
<tr>
<td><strong>FIP keepalive</strong></td>
<td>Periodic multicast FIP discovery advertisement sent from the FC switch or FCF to all ENodes to maintain connectivity.</td>
</tr>
<tr>
<td>Term</td>
<td>Definition</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| **FIP snooping**            | For VN_Port to VF_Port (VN2VF) paths (Technical Committee T1I BB-FC-5 specification), FIP snooping is a security feature enabled for FCoE VLANs on an Ethernet switch that connects ENodes to FC switches or FCFs. FIP snooping inspects data in FIP frames and uses that data to create firewall filters. The filters permit only traffic from sources that perform a successful FLOGI to the FC switch. All other traffic on the VLAN is denied. FIP snooping filters are installed on the ports in the FCoE VLAN.  
For VN_Port to VN_Port (VN2VN) paths (Technical Committee T1I BB-FC-6 specification), the FIP snooping security feature filters access between VN_Ports in a similar manner to VN2VF_Port FIP snooping.  
FIP snooping can also apply similarly to VE_Port to VE_Port (VE2VE) paths.  
FIP snooping can also snoop to provide additional visibility of FCoE Layer 2 operation.  
See also FCoE node (ENode). |
| **FIP snooping bridge**     | See FCoE transit switch and FIP snooping.                                                                                                                                                               |
| **host bus adapter (HBA)** | Physical mechanism that connects a host system to other FC network and storage devices. HBAs have a unique worldwide node name (WWNN) for the HBA node, which all of the ports on the HBA share, and each port on an HBA has a unique worldwide port name (WWPN). |
| **initiator**               | System component that originates an I/O command over an I/O bus or network. An FCoE server sending a request to an FC storage device is an example of an initiator.                                               |
| **iSCSI transit switch**   | Layer 2 Ethernet switch with a minimum set of best-practice Ethernet features to support iSCSI, along with optional enhancements. Minimum feature support is:  
• IEEE 802.3X asymmetric and symmetric flow control on ports not running in DCB mode  
• Priority-based flow control (PFC)  
• Enhanced transmission selection (ETS)  
• Data Center Bridging Capability Exchange Protocol (DCBX), including the iSCSI application TLV  
Other capabilities such as Internet storage name service (iSNS) are optional.                                                        |
| **interswitch link (ISL)**  | Link between the E_Ports of two FC switches in a common FC fabric. When two FCoE-based FC switches are connected together, there is a virtual ISL through Layer 2.                                            |
| **logout (LOGO)**           | For FC devices, an N_Port logs out from the FC network by sending an FC LOGO message to the F_Port of an FC switch. The switch can also send a LOGO message to an N_Port to terminate its connection.  
For FCoE devices, a VN_Port logs out from the FC network by sending a FIP LOGO message to the VF_Port of an FC switch. The switch can also send a LOGO message to a VN_Port to terminate its connection. |
### Table 9: Fibre Channel Terms (continued)

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>lossless Ethernet MAC</td>
<td>Full-duplex Ethernet MAC that implements Ethernet extensions to avoid Ethernet frame loss due to congestion and supports at least 2.5-KB jumbo frames. Each lossless Ethernet MAC combines with an FCoE Controller to perform FCoE termination functions on an ENode. See also priority-based flow control (PFC), quantized congestion notification (QCN), FCoE controller, and FCoE node (ENode).</td>
</tr>
<tr>
<td>lossless Ethernet network</td>
<td>Ethernet network composed of only full-duplex links and lossless Ethernet MACs and with CoS and flow control to prevent dropping of frames.</td>
</tr>
<tr>
<td>lossless transport</td>
<td>In DCB networks, the ability to switch FCoE frames over an Ethernet network without dropping any frames. Lossless transport uses mechanisms such as priority-based flow control and quantized congestion notification to control traffic flows and avoid congestion.</td>
</tr>
<tr>
<td>N_Port ID</td>
<td>See Fibre Channel ID (FCID).</td>
</tr>
<tr>
<td>N_Port ID virtualizer</td>
<td>Presents itself as an FC or FCoE switch to external devices, but connects to an actual FC or FCoE switch in the other direction to provide the FC-SW services.</td>
</tr>
<tr>
<td>N_Port ID virtualization (NPIV)</td>
<td>NPIV enables a physical N_Port to acquire multiple N_Port IDs. Each N_Port ID maps to a different application (such as a virtual machine) or to a different user. This allows you to associate one F_Port with many N_Port IDs and create multiple discrete, secure virtual links over one physical point-to-point connection. NPIV increases resource and bandwidth utilization and allows the implementation of access control, zoning, and port security on a per-application or per-user basis. After an N_Port performs a FLOGI and receives its first N_Port ID, it can request more N_Port IDs by sending FDISC messages. See also fabric login (FLOGI), fabric discovery (FDISC), and virtual link.</td>
</tr>
</tbody>
</table>

Copyright © 2014, Juniper Networks, Inc.
**Table 9: Fibre Channel Terms (continued)**

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>node port (N_Port)</td>
<td>N_Ports can be in two modes:</td>
</tr>
<tr>
<td></td>
<td>• Fabric N_Port—Node port that is an FC host or storage device end port in a point-to-point link between the device and the F_Port of an FC switch. The point-to-point link can be virtual or physical.</td>
</tr>
<tr>
<td></td>
<td>• Point-to-point N_Port—Node port that connects to another N_Port. The switch does not support this configuration.</td>
</tr>
<tr>
<td></td>
<td>N_Ports handle creation, detection, and flow of messages to and from the connected devices.</td>
</tr>
<tr>
<td>node worldwide name (WWNN)</td>
<td>WWN that is unique worldwide and is assigned to an FC node. An NWNN is valid for multiple ports that are on that node (this identifies the ports as network interfaces of a particular node).</td>
</tr>
<tr>
<td>port mode</td>
<td>Role that the port plays in the FC fabric (endpoint device, FC switch connection to endpoint devices, interswitch link).</td>
</tr>
<tr>
<td></td>
<td>See also node port (N_Port), virtual node port (VN_Port), proxy node port (NP_Port), fabric port (F_Port), and virtual fabric port (VF_Port).</td>
</tr>
<tr>
<td>port worldwide name (PWWN)</td>
<td>WWN that is unique worldwide and is assigned to an FC port.</td>
</tr>
<tr>
<td>priority-based flow control (PFC)</td>
<td>Link-level flow control mechanism defined by IEEE 802.1Qbb that allows independent flow control for each class of service (as defined in the 3-bit CoS field of the Ethernet header by IEEE 802.1Q tags) to ensure that no frame loss from congestion occurs in DCB networks.</td>
</tr>
<tr>
<td></td>
<td>PFC is an enhancement of the Ethernet PAUSE mechanism, but PFC controls classes of flows, whereas Ethernet PAUSE indiscriminately pauses all of the traffic on a link. With PFC, a receiving device can signal a transmitting device to pause transmission based on traffic class.</td>
</tr>
<tr>
<td></td>
<td>PFC provides application-specific bandwidth reservations so you can ensure that time-critical protocols and applications such as FCoE receive the priority necessary to prevent frame loss. PFC allows the same physical link to carry FCoE traffic and provide lossless service while also carrying loss-tolerant Ethernet traffic.</td>
</tr>
<tr>
<td></td>
<td>See also Ethernet PAUSE.</td>
</tr>
<tr>
<td>proxy gateway mode</td>
<td>Connects FCoE initiators to FC switches in a converged Ethernet and Fibre Channel network and acts as an intermediary for these devices. The FCoE-FC gateway represents and acts for the FCoE initiators in transactions from the FCoE initiators destined for an FC switch, including converting FIP and FCoE frames to FC frames. The gateway represents and acts for an FC switch in transactions from the FC switch destined for an FCoE initiator, including converting FC frames to FIP frames and encapsulating FC frames in Ethernet.</td>
</tr>
<tr>
<td>proxy node port (NP_Port)</td>
<td>N_Port on the QFX3500 switch that performs proxy functions when it is configured as an FCoE-FC gateway. The NP_Port acts as a proxy for the FCoE device VN_Ports in transactions with the FC switch.</td>
</tr>
</tbody>
</table>
Table 9: Fibre Channel Terms (continued)

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>quantized congestion notification</td>
<td>Mechanism defined by IEEE 802.1Qau that manages network congestion within a Layer 2 domain. When a queue reaches a configured threshold, QCN throttles traffic at the source of the congestion by transmitting messages that propagate back to the source and temporarily stop the source from transmitting. When the queue crosses the threshold that indicates the congestion has dissipated, QCN sends a message to allow the source to resume transmitting frames.</td>
</tr>
<tr>
<td>session</td>
<td>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.</td>
</tr>
<tr>
<td>server-provided MAC address</td>
<td>MAC address that an ENode assigns to one of its ENode MACs and is not assigned to any other ENode MAC in the same FCoE VLAN. An SPMA can be associated with more than one VN_Port at that ENode MAC.</td>
</tr>
<tr>
<td></td>
<td>The switch does not support SPMA.</td>
</tr>
<tr>
<td></td>
<td>See also ENode MAC and fabric-provided MAC address (FPMA).</td>
</tr>
<tr>
<td>storage area network (SAN)</td>
<td>Network whose primary purpose is the transfer of data between computer systems and storage devices. This term is most commonly used in the context of any network that supports block storage, usually iSCSI, FC, and FCoE networks.</td>
</tr>
<tr>
<td>target</td>
<td>System component that receives an I/O command. An FC storage device that receives a request from a server is an example of a target.</td>
</tr>
<tr>
<td>VE_Port</td>
<td>Virtual ports created to form a connection (an interswitch link) between two FCoE-based FC switches as part of a common FC fabric.</td>
</tr>
<tr>
<td>VE2VE (VE_Port to VE_Port)</td>
<td>The Fibre Channel Backbone - 5 (FC-BB-5) Rev 2.00 specification capability of FCFs to connect to each other as a single FCoE FC SAN.</td>
</tr>
<tr>
<td>VN2VF (VN_Port to VF_Port)</td>
<td>The Fibre Channel Backbone - 5 (FC-BB-5) Rev 2.00 specification capability of an ENode to connect to an FCF or to an FCoE-enabled FC SAN.</td>
</tr>
<tr>
<td>VN2VN (VN_Port to VN_Port)</td>
<td>The Fibre Channel Backbone - 6 (FC-BB-6) specification capability of an ENode to connect directly over Layer 2 to another ENode without the need of any FC-related services. This capability is most often used in small-scale FCoE SANs.</td>
</tr>
<tr>
<td>virtual fabric port (VF_Port)</td>
<td>Data-forwarding component that emulates an F_Port. A VF_Port is dynamically instantiated on successful completion of a FIP FLOGI exchange and connects to one or more VN_Ports. The term virtual indicates the use of a non-FC link such as an FCoE link.</td>
</tr>
<tr>
<td></td>
<td>See also fabric port (F_Port).</td>
</tr>
<tr>
<td>virtual link</td>
<td>Logical link connecting two FCoE Link End Points (LEPs) over a lossless Ethernet network, for example, the link between a VF_Port and a VN_Port. The MAC addresses of the two LEPs identifies a virtual link.</td>
</tr>
<tr>
<td></td>
<td>See also FCoE link end point (LEP) and lossless Ethernet network.</td>
</tr>
</tbody>
</table>
Table 9: Fibre Channel Terms (continued)

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>virtual node port (VN_Port)</td>
<td>Data-forwarding component that emulates an N_Port. With FCoE, a VN_Port is dynamically instantiated on successful completion of a FIP FLOGI exchange and connects to one or more VF_Ports. The term virtual indicates the use of a non-FC link such as an FCoE link. VN_Port is also used for the virtual N_Ports created in both FC and FCoE when additional NPIV-based logins occur over a previously created N_Port-to-VN_Port or N_Port-to-VF_Port connection. See also node port (N_Port).</td>
</tr>
<tr>
<td>well-known address (WKA)</td>
<td>Address identifier used to access a service provided by an FC fabric. The service can be distributed in many elements throughout a fabric, or it can be centralized in one element. A WKA is always accessible, regardless of zoning. An example of a WKA is the ALL-FCF-MACs address to which all FCFs listen.</td>
</tr>
<tr>
<td>worldwide name (WWN)</td>
<td>64-bit identifier that is similar to a MAC address except that it is not used for forwarding. It uniquely identifies an FC device. The WWN is derived from the IEEE organizationally unique identifier (OUI) and vendor-supplied information. A WWN is unique worldwide.</td>
</tr>
<tr>
<td>worldwide node name (WWNN)</td>
<td>See node worldwide name (NWWN).</td>
</tr>
<tr>
<td>worldwide port name (WWPN)</td>
<td>See port worldwide name (PWWN).</td>
</tr>
</tbody>
</table>

Related Documentation

- Overview of Fibre Channel on page 4
- Understanding QFabric System Terminology
CHAPTER 3

DCBX

- Understanding DCBX on page 87
- Understanding DCBX Application Protocol TLV Exchange on page 97

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 87
- DCBX Modes and Support on page 88
- DCBX Attribute Types on page 91
- DCBX Application Protocol TLV Exchange on page 92
- DCBX and PFC on page 93
- DCBX and ETS on page 94

DCBX Basics

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 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.
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 88
- Autonegotiation on page 90
- CNA Support for DCBX Modes on page 91
- Interface Support for DCBX on page 91

**DCBX Modes (Versions)**

The two most common DCBX modes are supported:
IEEE DCBX is 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 is 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 10 on page 89 summarizes the differences between IEEE DCBX and DCBX version 1.01, including show command output:

### Table 10: Summary of Differences Between IEEE DCBX and DCBX Version 1.01

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>IEEE DCBX</th>
<th>DCBX Version 1.01</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>OUI</strong></td>
<td>0x0080c2</td>
<td>0x001b21</td>
</tr>
<tr>
<td><strong>Frame Format</strong></td>
<td>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.</td>
<td>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.</td>
</tr>
<tr>
<td><strong>Symmetric/asymmetric configuration with peer</strong></td>
<td>Asymmetric or symmetric</td>
<td>Symmetric only</td>
</tr>
<tr>
<td><strong>Differences in the show dcbx interface interface-name operational command</strong></td>
<td>• 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.</td>
<td>• 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).</td>
</tr>
</tbody>
</table>
For more information about how each DCBX mode exchanges TLVs, see the following specifications:

- For DCBX version 1.01—

**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 92
• Symmetric Attributes on page 92
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 92
- FCoE Application Protocol TLV Exchange on page 93
- Disabling Application Protocol TLV Exchange on page 93

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 94
- ETS Advertisement and Peer Configuration on page 94
- ETS Recommendation TLV on page 95

**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 97
- Understanding DCB Features and Requirements on page 16
- Understanding CoS Flow Control (Ethernet PAUSE and PFC) on page 66
- *Understanding CoS Hierarchical Port Scheduling (ETS)*
- Understanding FCoE on page 20
- Configuring the DCBX Mode on page 185
- Configuring DCBX Autonegotiation on page 186
- Disabling the ETS Recommendation TLV on page 189
- Example: Configuring DCBX Application Protocol TLV Exchange on page 105
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 97) 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 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 (see “Application Maps” on page 98) and 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.

This topic describes:

- Applications on page 97
- Application Maps on page 98
- Classifying and Prioritizing Application Traffic on page 99
- Enabling Interfaces to Exchange Application Protocol Information on page 100
- Disabling DCBX Application Protocol Exchange on page 100

Applications

Before an interface can exchange application protocol information, you need to define the applications that you want to advertise, except FCoE if FCoE is the only application that you want the interface to advertise.
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 87
- Understanding CoS Classifiers
- Configuring DCBX Autonegotiation on page 186
- Disabling the ETS Recommendation TLV on page 189
- Defining an Application for DCBX Application Protocol TLV Exchange on page 189
- Configuring an Application Map for DCBX Application Protocol TLV Exchange on page 191
- Applying an Application Map to an Interface for DCBX Application Protocol TLV Exchange on page 192
- Example: Configuring DCBX Application Protocol TLV Exchange on page 105
• Example: Configuring Unicast Classifiers
PART 2

Configuration

- Configuration Examples on page 105
- Configuration Examples (ELS CLI for Platforms that Support FCoE Only) on page 125
- Configuration Tasks (Fibre Channel, FCoE, FIP, and FIP Snooping) on page 169
- Configuration Tasks (DCBX) on page 185
- Configuration Statements on page 193
- Configuration Statements (ELS CLI for Platforms that Support FCoE Only) on page 219
CHAPTER 4

Configuration Examples

- Example: Configuring DCBX Application Protocol TLV Exchange on page 105
- Example: Configuring CoS PFC for FCoE Traffic on page 115

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 106
- Overview on page 106
- Configuration on page 109
- Verification on page 111

## 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 186 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 11 on page 107 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 12 on page 107 shows the default untrusted classifier IEEE 802.1 code-point values to unicast forwarding class mapping for ports in access mode.

Table 11: Default IEEE 802.1 Classifiers for Trunk Ports and Tagged-Access Ports (Default Trusted Classifier)

<table>
<thead>
<tr>
<th>Code Point</th>
<th>Forwarding Class</th>
<th>Loss Priority</th>
</tr>
</thead>
<tbody>
<tr>
<td>be (000)</td>
<td>best-effort</td>
<td>low</td>
</tr>
<tr>
<td>be1 (001)</td>
<td>best-effort</td>
<td>low</td>
</tr>
<tr>
<td>ef (010)</td>
<td>best-effort</td>
<td>low</td>
</tr>
<tr>
<td>ef1 (011)</td>
<td>fcoe</td>
<td>low</td>
</tr>
<tr>
<td>af1 (100)</td>
<td>no-loss</td>
<td>low</td>
</tr>
<tr>
<td>af12 (101)</td>
<td>best-effort</td>
<td>low</td>
</tr>
<tr>
<td>nc1 (110)</td>
<td>network-control</td>
<td>low</td>
</tr>
<tr>
<td>nc2 (111)</td>
<td>network-control</td>
<td>low</td>
</tr>
</tbody>
</table>

Table 12: Default IEEE 802.1 Unicast Classifiers for Access Ports (Default Untrusted Classifier)

<table>
<thead>
<tr>
<th>Code Point</th>
<th>Forwarding Class</th>
<th>Loss Priority</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>best-effort</td>
<td>low</td>
</tr>
<tr>
<td>001</td>
<td>best-effort</td>
<td>low</td>
</tr>
<tr>
<td>010</td>
<td>best-effort</td>
<td>low</td>
</tr>
<tr>
<td>011</td>
<td>best-effort</td>
<td>low</td>
</tr>
<tr>
<td>100</td>
<td>best-effort</td>
<td>low</td>
</tr>
<tr>
<td>101</td>
<td>best-effort</td>
<td>low</td>
</tr>
<tr>
<td>110</td>
<td>best-effort</td>
<td>low</td>
</tr>
<tr>
<td>111</td>
<td>best-effort</td>
<td>low</td>
</tr>
</tbody>
</table>
### 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 13 on page 108 shows the configuration components for this example.

**Table 13: Components of DCBX Application Protocol Exchange**

<table>
<thead>
<tr>
<th>Component</th>
<th>Settings</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hardware</td>
<td>QFX Series device</td>
</tr>
<tr>
<td>LLDP</td>
<td>Enabled by default on Ethernet interfaces</td>
</tr>
<tr>
<td>DCBX</td>
<td>Enabled by default on Ethernet interfaces</td>
</tr>
<tr>
<td>iSCSI application (Layer 4)</td>
<td>Application name—<code>iscsi</code></td>
</tr>
<tr>
<td></td>
<td>protocol—<code>TCP</code></td>
</tr>
<tr>
<td></td>
<td>destination-port—<code>3260</code></td>
</tr>
<tr>
<td></td>
<td>code-points—<code>111</code></td>
</tr>
<tr>
<td>PTP application (Layer 2)</td>
<td>Application name—<code>ptp</code></td>
</tr>
<tr>
<td></td>
<td>ether-type—<code>0x88F7</code></td>
</tr>
<tr>
<td></td>
<td>code-points—<code>001, 101</code></td>
</tr>
<tr>
<td>FCoE application (Layer 2)</td>
<td>Application name—<code>fcoe</code></td>
</tr>
<tr>
<td></td>
<td>ether-type—<code>0x8906</code></td>
</tr>
<tr>
<td></td>
<td>code-points—<code>011</code></td>
</tr>
<tr>
<td></td>
<td><strong>NOTE:</strong> 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.</td>
</tr>
<tr>
<td>Application maps</td>
<td><code>dcbx-iscsi-fcoe-app-map</code>—Maps the iSCSI and FCoE applications to IEEE 802.1p code points</td>
</tr>
<tr>
<td></td>
<td><code>dcbx-iscsi-tpm-app-map</code>—Maps iSCSI and PTP applications to IEEE 802.1p code points</td>
</tr>
</tbody>
</table>
Table 13: Components of DCBX Application Protocol Exchange Configuration Topology (continued)

<table>
<thead>
<tr>
<th>Component</th>
<th>Settings</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interfaces</td>
<td>xe-0/0/10—Configured to exchange FCoE and iSCSI application TLVs (uses application map <code>dcbx-iscsi-fcoe-app-map</code>, 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 <code>dcbx-iscsi-tp-p-app-map</code>)</td>
</tr>
<tr>
<td>PFC congestion notification profile for FCoE application exchange</td>
<td>fcoe-cnp:</td>
</tr>
<tr>
<td></td>
<td>• Code point—011</td>
</tr>
<tr>
<td></td>
<td>• Interface—xe-0/0/10</td>
</tr>
<tr>
<td>Behavior aggregate classifiers (map forwarding classes to incoming packets by the packet's IEEE 802.1 code point)</td>
<td>fcoe-iscsi-cl1:</td>
</tr>
<tr>
<td></td>
<td>• Maps the <code>fcoe</code> forwarding class to the IEEE 802.1p code point used for the FCoE application (011) and a loss priority of high</td>
</tr>
<tr>
<td></td>
<td>• Maps the <code>network-control</code> forwarding class to the IEEE 802.1p code point used for the iSCSI application (111) and a loss priority of high</td>
</tr>
<tr>
<td></td>
<td>• Applied to interface xe-0/0/10</td>
</tr>
<tr>
<td></td>
<td>iscsi-tp-tp-cl2:</td>
</tr>
<tr>
<td></td>
<td>• Maps the <code>network-control</code> forwarding class to the IEEE 802.1p code point used for the iSCSI application (111) and a loss priority of low</td>
</tr>
<tr>
<td></td>
<td>• Maps the <code>best-effort</code> forwarding class to the IEEE 802.1p code points used for the PTP application (001 and 101) and a loss priority of low</td>
</tr>
<tr>
<td></td>
<td>• Applied to interface xe-0/0/11</td>
</tr>
</tbody>
</table>

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.

```plaintext
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-pt-p-app-map application PTP code-points [001, 101]
```
set protocols dcbbx interface xe-0/0/10 application-map dcbx-iscsi-fcoe-app-map
set protocols dcbbx interface xe-0/0/11 application-map dcbx-iscsi-ptp-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-ptp-cl2 import default forwarding-class
network-control loss-priority low code-points 111
set class-of-service classifiers ieee-802.1 iscsi-ptp-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
   user@switch# set application-maps dcbx-iscsi-fcoe-app-map application FCoE code-points

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
   user@switch# set application-maps dcbx-iscsi-ptp-app-map application PTP code-points

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 dcbbx]
   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 111
- Verifying the Application Map Configuration on page 112
- Verifying DCBX Application Protocol Exchange Interface Configuration on page 112
- Verifying the PFC Configuration on page 113
- Verifying the Classifier Configuration on page 114

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-tpp-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-tpp-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-tpp-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-tpp-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`:

```plaintext
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-ptp-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-ptp-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-ptp-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-tp-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-tp-cl2` is mapped to interface `xe-0/0/11.0`.

### Related Documentation

- **Example: Configuring Unicast Classifiers**
  - Defining an Application for DCBX Application Protocol TLV Exchange on page 189
  - Configuring an Application Map for DCBX Application Protocol TLV Exchange on page 191
  - Applying an Application Map to an Interface for DCBX Application Protocol TLV Exchange on page 192
  - Configuring DCBX Autonegotiation on page 186
  - `show dcbx on page 248`
  - `show dcbx neighbors on page 249`
  - Understanding DCBX Application Protocol TLV Exchange on page 97
  - Using DCBX Protocol to Lower Costs

### 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 flag 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.
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.
- Configure a congestion notification profile to apply PFC to the FCoE traffic.
- Apply the classifier and the PFC configuration to ingress interfaces.
- Configure the bandwidth scheduling for the FCoE forwarding class output queue.
- 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).
- Configure the bandwidth scheduling for the FCoE priority group.
- Apply the scheduling to the egress interfaces.

**NOTE:** If you are using Junos OS Release 12.2 or later, use the default forwarding classes for the lossless fcoe forwarding class. If you explicitly configure default lossless forwarding classes, the traffic mapped to those forwarding classes is treated as lossy (best-effort) traffic and does not receive lossless treatment.

In Junos OS Release 12.3 and later, you can include the no-loss packet drop attribute in explicit forwarding class configurations to configure a lossless forwarding class.

This example describes how to configure PFC for FCoE traffic:

- **Requirements** on page 116
- **Overview** on page 116
- **Configuration** on page 118
- **Verification** on page 121

### Requirements

This example uses the following hardware and software components:

- A Juniper Networks QFX3500 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:

- Assign FCoE traffic to the FCoE priority at the ingress.
- Create and apply CoS for the FCoE traffic using ETS (hierarchical port scheduling).
• Apply PFC to the FCoE traffic.
• Apply the configuration to ingress and egress 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.

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

**Topology**

Table 14 on page 117 shows the configuration components for this example.

**Table 14: Components of the PFC for FCoE Traffic Configuration Topology**

<table>
<thead>
<tr>
<th>Component</th>
<th>Settings</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hardware</td>
<td>QFX3500 switch</td>
</tr>
<tr>
<td>Behavior aggregate classifier (maps the FCoE forwarding class to incoming packets by IEEE 802.1 code point)</td>
<td>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</td>
</tr>
<tr>
<td>PFC congestion notification profile</td>
<td>fcoe-cnp: Code point 011 Ingress interfaces: xe-0/0/31, xe-0/0/32, xe-0/0/33, xe-0/0/34</td>
</tr>
<tr>
<td>FCoE queue scheduler</td>
<td>fcoe-sched: Minimum bandwidth 3g Maximum bandwidth 100% Priority low</td>
</tr>
<tr>
<td>Forwarding class-to-scheduler mapping</td>
<td>Scheduler map fcoe-map: Forwarding class fcoe Scheduler fcoe-sched</td>
</tr>
<tr>
<td>Forwarding class set (FCoE priority group)</td>
<td>fcoe-pg: Forwarding class fcoe Egress interfaces: xe-0/0/31, xe-0/0/32, xe-0/0/33, xe-0/0/34</td>
</tr>
<tr>
<td>Traffic control profile</td>
<td>fcoe-tcp: Scheduler map fcoe-map Minimum bandwidth 3g Maximum bandwidth 100%</td>
</tr>
</tbody>
</table>

Figure 6 on page 118 shows a block diagram of the configuration components and the configuration flow of the CLI statements used in the example.
Configure

**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:

```
[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-0/0/33 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-0/0/33 congestion-notification-profile fcoe-cnp
set interfaces xe-0/0/34 congestion-notification-profile fcoe-cnp
set interfaces xe-0/0/31 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
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-0/0/33 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
```

**Step-by-Step Procedure**
To configure the FCoE forwarding class (priority), ingress classifier, output queue scheduling, forwarding class set (priority group) and its output port scheduling, PFC application, and interfaces to set up PFC for FCoE traffic:

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:
3. Apply the PFC configuration to the ingress interfaces:

```
[edit class-of-service]
user@switch# set congestion-notification-profile fcoe-cnp input ieee-802.1 code-point 01 pfc
```

4. Assign the classifier 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
```

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
```

7. Configure the forwarding class set for the FCoE traffic:

```
[edit class-of-service]
user@switch# set forwarding-class-sets fcoe-pg class fcoe
```

8. 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
```

9. 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
```

## 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):

```
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 122](#)
- [Verifying the Ingress Interface PFC Configuration on page 122](#)
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

<table>
<thead>
<tr>
<th>Priority</th>
<th>PFC</th>
<th>MRU</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>Disabled</td>
<td></td>
</tr>
<tr>
<td>001</td>
<td>Disabled</td>
<td></td>
</tr>
<tr>
<td>010</td>
<td>Disabled</td>
<td></td>
</tr>
<tr>
<td>011</td>
<td>Enabled</td>
<td>2500</td>
</tr>
<tr>
<td>100</td>
<td>Disabled</td>
<td></td>
</tr>
<tr>
<td>101</td>
<td>Disabled</td>
<td></td>
</tr>
<tr>
<td>110</td>
<td>Disabled</td>
<td></td>
</tr>
<tr>
<td>111</td>
<td>Disabled</td>
<td></td>
</tr>
</tbody>
</table>

Type: Output

<table>
<thead>
<tr>
<th>Priority</th>
<th>Flow-Control-Queues</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>0</td>
</tr>
<tr>
<td>001</td>
<td>1</td>
</tr>
<tr>
<td>010</td>
<td>2</td>
</tr>
<tr>
<td>011</td>
<td>3</td>
</tr>
<tr>
<td>100</td>
<td>4</td>
</tr>
<tr>
<td>101</td>
<td>5</td>
</tr>
<tr>
<td>110</td>
<td>6</td>
</tr>
<tr>
<td>111</td>
<td>7</td>
</tr>
</tbody>
</table>
```

**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`. 

---

Copyright © 2014, Juniper Networks, Inc.
**Action**  
List the ingress interfaces using the operational mode command `show configuration class-of-service interfaces`:

```plaintext
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)
- Overview of CoS Changes Introduced in Junos OS Release 12.2
- Understanding CoS Flow Control (Ethernet PAUSE and PFC) on page 66
CHAPTER 5

Configuration Examples (ELS CLI for Platforms that Support FCoE Only)

- Example: Configuring CoS for FCoE Transit Switch Traffic Across an MC-LAG on page 125
- Example: Configuring VN2VN_Port FIP Snooping (FCoE Hosts Directly Connected to the Same FCoE Transit Switch) on page 147
- Example: Configuring VN2VN_Port FIP Snooping (FCoE Hosts Directly Connected to Different FCoE Transit Switches) on page 152
- Example: Configuring VN2VN_Port FIP Snooping (FCoE Hosts Indirectly Connected Through an Aggregation Layer FCoE Transit Switch) on page 159

**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 Fiber 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.

QFX3500 and QFX3600 Virtual Chassis switches do not support FCoE.

This topic describes:

- Requirements on page 126
- Overview on page 126
- Configuration on page 130
- Verification on page 139

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) and enhanced transmission selection (ETS; hierarchical scheduling of resources for the FCoE forwarding class priority and for the forwarding class set priority group).

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 7 on page 127.

**Figure 7: Supported Topology for an MC-LAG on an FCoE Transit Switch**

Table 15 on page 127 shows the configuration components for this example.

**Table 15: Components of the CoS for FCoE Traffic Across an MC-LAG Configuration Topology**

<table>
<thead>
<tr>
<th>Component</th>
<th>Settings</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hardware</td>
<td>Four QFX5100 switches running the ELS CLI (two to form the MC-LAG as passthrough transit switches and two transit switches for FCoE access).</td>
</tr>
<tr>
<td>Forwarding class (all switches)</td>
<td>Default <strong>fcOE</strong> forwarding class.</td>
</tr>
<tr>
<td>Classifier (forwarding class mapping of incoming traffic to IEEE priority)</td>
<td>Default IEEE 802.1p trusted classifier on all FCoE interfaces.</td>
</tr>
</tbody>
</table>
### Table 15: Components of the CoS for FCoE Traffic Across an MC-LAG Configuration Topology (continued)

<table>
<thead>
<tr>
<th>Component</th>
<th>Settings</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>LAGs and MC-LAG</strong></td>
<td>S1—Ports xe-0/0/10 and xe-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. S2—Ports xe-0/0/10 and xe-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. NOTE: Ports xe-0/0/20 and xe-0/0/21 on Switches S1 and S2 are the members of the MC-LAG. TS1—Ports xe-0/0/25 and xe-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. TS2—Ports xe-0/0/25 and xe-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.</td>
</tr>
<tr>
<td><strong>FCoE queue scheduler (all switches)</strong></td>
<td>fcoe-sched: Minimum bandwidth 3g Maximum bandwidth 100% Priority low</td>
</tr>
<tr>
<td><strong>Forwarding class-to-scheduler mapping (all switches)</strong></td>
<td>Scheduler map fcoe-map: Forwarding class fcoe Scheduler fcoe-sched</td>
</tr>
<tr>
<td><strong>Forwarding class (FCoE priority group, all switches)</strong></td>
<td>fcoe-pg: Forwarding class fcoe Egress interfaces: 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</td>
</tr>
<tr>
<td><strong>Traffic control profile (all switches)</strong></td>
<td>fcoe-tcp: Scheduler map fcoe-map Minimum bandwidth 3g Maximum bandwidth 100%</td>
</tr>
</tbody>
</table>
Table 15: Components of the CoS for FCoE Traffic Across an MC-LAG Configuration Topology (continued)

<table>
<thead>
<tr>
<th>Component</th>
<th>Settings</th>
</tr>
</thead>
</table>
| PFC congestion notification profile (all switches) | **fcoe-cnp:**  
  Code point 011  
  Ingress interfaces:  
  • 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 |
| FCoE VLAN name and tag ID | Name—**fcoe_vlan**  
  ID—100  
  Include the FCoE VLAN on the interfaces that carry FCoE traffic on all four switches. |
| FIP snooping | 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.  
  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. |

**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 enhanced transmission selection (ETS, also known as hierarchical scheduling) 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.

• Apply the ETS scheduling to the interfaces.

• Configure the interface mode, MTU, and FCoE trusted or untrusted state for each interface to support lossless FCoE transport.

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:

• Configuring MC-LAG Switches S1 and S2 on page 132
• Configuring FCoE Transit Switches TS1 and TS2 on page 134
• Results on page 136

CLI Quick Configuration

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.
Switch S1 and Switch S2

```plaintext
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 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
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
```

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.

Switch TS1 and Switch TS2

```plaintext
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 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
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
```

Copyright © 2014, Juniper Networks, Inc.
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

Configuring MC-LAG Switches S1 and S2

Step-by-Step Procedure
To configure CoS resource scheduling (ETS), PFC, the FCoE VLAN, and the 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, so you do not configure them):

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. 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
   ```

4. 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
   ```

5. 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
   ```
6. 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
```

7. 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
```

8. Configure the VLAN for FCoE traffic (fcoe_vlan):

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

9. 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
```

10. 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
```

11. 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
```

12. 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
```

13. 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
```
Configuring FCoE Transit Switches TS1 and TS2

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 resource scheduling (ETS), PFC, the FCoE VLAN, and the 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):

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. 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
   ```

4. 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
   ```

5. 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
   ```

6. 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
   ```

7. 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

8. Configure the VLAN for FCoE traffic (fcoe_vlan):
   [edit vlans]
   user@switch# set fcoe_vlan vlan-id 100

9. 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

10. 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

11. 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

12. 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

13. 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.

14. 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
```

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):

```
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 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.

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):

```bash
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 01 {
          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 139
- Verifying That the Priority Group Output Scheduler (Traffic Control Profile) Has Been Created on page 140
- Verifying That the Forwarding Class Set (Priority Group) Has Been Created on page 140
- Verifying That Priority-Based Flow Control Has Been Enabled on page 141
- Verifying That the Interface Class of Service Configuration Has Been Created on page 142
- Verifying That the Interfaces Are Correctly Configured on page 143
- Verifying That FIP Snooping Is Enabled on the FCoE VLAN on FCoE Transit Switches TS1 and TS2 Access Interfaces on page 146
- Verifying That the FIP Snooping Mode Is Correct on FCoE Transit Switches TS1 and TS2 on page 147

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

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

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

<table>
<thead>
<tr>
<th>Forwarding class</th>
<th>Index</th>
</tr>
</thead>
<tbody>
<tr>
<td>fcoe</td>
<td>1</td>
</tr>
</tbody>
</table>
```

**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
<table>
<thead>
<tr>
<th>Priority</th>
<th>PFC</th>
<th>MRU</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>Disabled</td>
<td></td>
</tr>
<tr>
<td>001</td>
<td>Disabled</td>
<td></td>
</tr>
<tr>
<td>010</td>
<td>Disabled</td>
<td></td>
</tr>
<tr>
<td>011</td>
<td>Enabled</td>
<td>2500</td>
</tr>
<tr>
<td>100</td>
<td>Disabled</td>
<td></td>
</tr>
<tr>
<td>101</td>
<td>Disabled</td>
<td></td>
</tr>
<tr>
<td>110</td>
<td>Disabled</td>
<td></td>
</tr>
<tr>
<td>111</td>
<td>Disabled</td>
<td></td>
</tr>
</tbody>
</table>

Type: Output
<table>
<thead>
<tr>
<th>Priority</th>
<th>Flow-Control-Queues</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>0</td>
</tr>
<tr>
<td>001</td>
<td>1</td>
</tr>
<tr>
<td>010</td>
<td>2</td>
</tr>
<tr>
<td>011</td>
<td>3</td>
</tr>
<tr>
<td>100</td>
<td>4</td>
</tr>
<tr>
<td>101</td>
<td>5</td>
</tr>
<tr>
<td>110</td>
<td>6</td>
</tr>
<tr>
<td>111</td>
<td>7</td>
</tr>
</tbody>
</table>
```

**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.

**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;
}
```
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;
  }
}
```
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`:

```plaintext
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 900000 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 115*
- *Example: Configuring CoS Hierarchical Port Scheduling (ETS)*
- *Understanding Multichassis Link Aggregation*
- *Understanding MC-LAGs on an FCoE Transit Switch on page 62*

**Example: Configuring VN2VN_Port FIP Snooping (FCoE Hosts Directly Connected to the Same FCoE Transit Switch)**

This example shows how to configure VN_Port to VN_Port (VN2VN_Port) FIP snooping when the hosts are directly connected to the same FCoE transit switch.
VN2VN_Port FIP snooping on an FCoE transit switch provides security to help prevent unauthorized access and data transmission on a bridge that connects ENodes in the Ethernet network. VN2VN_Port FIP snooping provides security for virtual links by creating filters based on information gathered (snooped) about FCoE devices during FIP transactions.

VN2VN_Port FIP snooping is conceptually similar to VN2VN_Port FIP snooping between VN_Ports and VF_Ports, but VN2VN_Port FIP snooping does not require traffic between VN_Ports to traverse the Fibre Channel (FC) switch or FCoE forwarder (FCF). Instead, a VN_Port communicates transparently through the transit switch on a virtual link that emulates a direct connection to the VN_Port at the other end of the virtual link.

To configure VN2VN_Port FIP snooping when the hosts are directly connected to the same FCoE transit switch, you must follow these configuration rules:

- VN2VN_Port traffic must use a dedicated FCoE VLAN, and all ENodes that communicate using VN2VN_Port FIP snooping must use that FCoE VLAN. You cannot mix VN2VN_Port FIP snooping traffic with VN2VF_Port FIP snooping traffic in the same FCoE VLAN.

- ENode-facing ports must be set in trunk interface mode.
- ENode-facing ports must be untrusted ports.
- Network-facing (switch-facing) ports must be set in trunk interface mode.
- Network-facing ports must be FCoE trusted ports.
- Explicitly configure the beacon period. The beacon period is essentially a keepalive timer for virtual link maintenance.

When you enable VN2VF_Port FIP snooping, the system snoops VN_Port to VF_Port packets and enforces security only on VN_Port to VF_Port virtual links. When you enable VN2VN_Port FIP snooping, the system snoops VN_Port to VN_Port packets and enforces security only on VN_Port to VN_Port virtual links.

The transit switch applies VN2VN_Port FIP snooping filters at the ports associated with the FCoE VLANs on which you enable VN2VF FIP snooping.
This example describes how to configure VN2VN_Port FIP snooping when the FCoE hosts are directly connected to the same transit switch:

- Requirements on page 149
- Overview on page 149
- Configuration on page 150
- Verification on page 151

Requirements

This example uses the following hardware and software components:

- One Juniper Networks QFX5100 Switch running the ELS CLI and used as a transit switch
- Junos OS Release 13.2 or later for the QFX Series
- Two FCoE hosts that have ENodes

Overview

This example shows you how to:

- Set the correct interface mode on the transit switch.
- Configure the interfaces to use the dedicated FCoE VLAN for VN2VN_Port FIP snooping.
- Configure the dedicated FCoE VLAN for VN2VN_Port FIP snooping traffic.
- Enable VN2VN_Port FIP snooping on the FCoE VLAN and configure the beacon period.

Topology

Table 16 on page 149 shows the configuration components for this example.

Table 16: Components of the VN2VN_Port FIP Snooping Configuration Topology (FCoE Hosts Directly Connected to the Same FCoE Transit Switch)

<table>
<thead>
<tr>
<th>Component</th>
<th>Settings</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hardware</td>
<td>QFX5100 switch running the ELS CLI (FCoE transit switch TS1)</td>
</tr>
<tr>
<td></td>
<td>Two FCoE hosts that have ENodes (ENode1 and ENode2, respectively)</td>
</tr>
<tr>
<td>Interfaces and interface mode</td>
<td>• Interface xe-0/0/20, interface mode trunk, connects directly to the FCoE host with ENode1.</td>
</tr>
<tr>
<td></td>
<td>• Interface xe-0/0/21, interface mode trunk, connects directly to the FCoE host with ENode2.</td>
</tr>
<tr>
<td>Interface VLAN membership</td>
<td>Both interfaces use VLAN vlan200.</td>
</tr>
<tr>
<td>VN2VN_Port FIP snooping VLAN</td>
<td>VLAN name—vlan200</td>
</tr>
<tr>
<td></td>
<td>VLAN ID—200</td>
</tr>
<tr>
<td>FIP snooping mode and beacon period</td>
<td>Set examine-vn2vn (VN2VN_Port FIP snooping)</td>
</tr>
<tr>
<td></td>
<td>Beacon period—90000 ms</td>
</tr>
</tbody>
</table>
Figure 8 on page 150 shows the network topology for this example.

Figure 8: VN2VN_Port FIP Snooping (FCoE Hosts Connected to Same Transit Switch) Topology

Configuration

<table>
<thead>
<tr>
<th>CLI Quick Configuration</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>To quickly configure VN2VN_Port FIP snooping for FCoE hosts connected directly to the same transit switch, 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:</td>
<td></td>
</tr>
<tr>
<td>set interfaces xe-0/0/20 unit 0 family ethernet-switching interface-mode trunk</td>
<td></td>
</tr>
<tr>
<td>set interfaces xe-0/0/21 unit 0 family ethernet-switching interface-mode trunk</td>
<td></td>
</tr>
<tr>
<td>set interfaces xe-0/0/20 unit 0 family ethernet-switching vlan members vlan200</td>
<td></td>
</tr>
<tr>
<td>set interfaces xe-0/0/21 unit 0 family ethernet-switching vlan members vlan200</td>
<td></td>
</tr>
<tr>
<td>set vlans vlan200 vlan-id 200</td>
<td></td>
</tr>
<tr>
<td>set vlans vlan200 forwarding-options fip-security examine-vn2vn beacon-period 90000</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step-by-Step Procedure</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>To configure interface mode, configure interface VLAN membership in the FCoE VLAN dedicated to VN2VN_Port traffic, configure the VLAN, set the beacon period, and enable VN2VN_Port FIP snooping:</td>
<td></td>
</tr>
<tr>
<td>1. Configure the modes of the interfaces that connect directly to the FCoE host ENodes:</td>
<td></td>
</tr>
<tr>
<td>user@switch# set interfaces xe-0/0/20 unit 0 family ethernet-switching interface-mode trunk</td>
<td></td>
</tr>
<tr>
<td>set interfaces xe-0/0/21 unit 0 family ethernet-switching interface-mode trunk</td>
<td></td>
</tr>
<tr>
<td>2. Configure the interface VLAN membership so that the interfaces connected to the ENodes are members of the dedicated VN2VN_Port VLAN (vlan200):</td>
<td></td>
</tr>
<tr>
<td>user@switch# set interfaces xe-0/0/20 unit 0 family ethernet-switching vlan members vlan200</td>
<td></td>
</tr>
<tr>
<td>set interfaces xe-0/0/21 unit 0 family ethernet-switching vlan members vlan200</td>
<td></td>
</tr>
<tr>
<td>3. Configure the FCoE VLAN dedicated to VN2VN_Port FIP snooping:</td>
<td></td>
</tr>
<tr>
<td>user@switch# set vlans vlan200 vlan-id 200</td>
<td></td>
</tr>
<tr>
<td>4. Enable VN2VN_Port FIP snooping on the VLAN and configure the beacon period:</td>
<td></td>
</tr>
<tr>
<td>user@switch# set vlans vlan200 forwarding-options fip-security examine-vn2vn beacon-period 90000</td>
<td></td>
</tr>
</tbody>
</table>
Verification

To verify that the VN2VN_Port FIP snooping configuration has been created and is operating properly, perform these tasks:

- Verifying That VN2VN_Port FIP Snooping is Enabled on the FCoE VLAN on page 151

Verifying That VN2VN_Port FIP Snooping is Enabled on the FCoE VLAN

Purpose

Verify that VN2VN_Port FIP snooping is enabled on the correct VLAN (vlan200), the beacon period is set to 90000 milliseconds, and the correct interfaces (xe-0/0/20 and xe-0/0/21) are members of the VLAN.

Action

List the FIP snooping information using the operational mode command `show fip snooping detail`.

```
user@switch> show fip snooping detail
VLAN: vlan200,  Mode: VN2VN Snooping
   FC-MAP: 0e:fd:00
Beacon_Period:  90000
VN2VN Mode: Point-to-Point
   Enode Information
      Enode-MAC: 10:10:94:01:00:02,       Interface: xe-0/0/20
      Active VN_Ports : 1
         VN_Port Information
         VN-Port MAC: 0e:fd:00:00:0a:01
         Active Sessions : 1
         Session Information
         Vlink far-end VN-Port-MAC: 0e:fd:00:00:0b:01
      Enode-MAC: 10:10:94:01:00:02,       Interface: xe-0/0/21
      Active VN_Ports : 1
         VN_Port Information
         VN-Port MAC: 0e:fd:00:00:0b:01
         Active Sessions : 1
         Session Information
         Vlink far-end VN-Port-MAC: 0e:fd:00:00:0a:01
```

Meaning

The `show fip snooping detail` command lists all of the transit switch information about VN2VN_Port FIP snooping and VN2VF_Port FIP snooping. The command shows that:

- The VLAN is `vlan200`.
- The mode is FIP snooping mode `VN2VN`, for VN2VN_Port FIP snooping. (If the Mode field shows `VN2VF`, then the FIP snooping mode is VN2VF_Port FIP snooping.)
- The beacon period is `90000`.
- The interfaces for the ENodes are `xe-0/0/20` and `xe-0/0/21`.

In addition, this useful command shows information about the ENodes and the VN2VN_Port sessions.

Related Documentation

- Example: Configuring VN2VN_Port FIP Snooping (FCoE Hosts Directly Connected to Different FCoE Transit Switches) on page 152
Example: Configuring VN2VN_Port FIP Snooping (FCoE Hosts Indirectly Connected Through an Aggregation Layer FCoE Transit Switch) on page 159

Example: Configuring VN2VN_Port FIP Snooping (FCoE Hosts Directly Connected to Different FCoE Transit Switches) on page 159

Example: Enabling VN2VN_Port FIP Snooping and Configuring the Beacon Period on an FCoE Transit Switch on page 182

Example: Understanding VN_Port to VN_Port FIP Snooping on an FCoE Transit Switch on page 45

Example: Configuring VN2VN_Port FIP Snooping (FCoE Hosts Directly Connected to Different FCoE Transit Switches)

This example shows how to configure VN_Port to VN_Port (VN2VN_Port) FIP snooping when the hosts are directly connected to different FCoE transit switches, and the transit switches are directly connected to each other.

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 VN2VN_Port FIP Snooping (FCoE Hosts Directly Connected to Different FCoE Transit Switches). For ELS details, see Getting Started with Enhanced Layer 2 Software.

VN2VN_Port FIP snooping on an FCoE transit switch provides security to help prevent unauthorized access and data transmission on a bridge that connects ENodes in the Ethernet network. VN2VN_Port FIP snooping provides security for virtual links by creating filters based on information gathered (snooped) about FCoE devices during FIP transactions.

VN2VN_Port FIP snooping is conceptually similar to VN2VF_Port FIP snooping between VN_Ports and VF_Ports, but VN2VN_Port FIP snooping does not require traffic between VN_Ports to traverse the Fibre Channel (FC) switch or FCoE forwarder (FCF). Instead, a VN_Port communicates transparently through one or more transit switches on a virtual link that emulates a direct connection to the VN_Port at the other end of the virtual link.
To configure VN2VN_Port FIP snooping when the hosts are directly connected to different FCoE transit switches, and the transit switches are directly connected to each other, you must follow these configuration rules:

- VN2VN_Port traffic must use a dedicated FCoE VLAN, and all ENodes that communicate using VN2VN_Port FIP snooping must use that FCoE VLAN. The FCoE VLAN must be configured on each transit switch. You cannot mix VN2VN_Port FIP snooping traffic with VN2VF_Port FIP snooping traffic in the same FCoE VLAN.

**NOTE:** An FCoE VLAN can support either VN2VF_Port FIP snooping or VN2VN_Port FIP snooping, but not both. Configure separate FCoE VLANs for VN2VF_Port FIP snooping traffic and for VN2VN_Port FIP snooping traffic. On FCoE VLANs that are configured as VN2VN_Port FIP snooping VLANs, VN2VF_Port traffic is dropped.

- ENode-facing ports must be set in **trunk** interface mode.
- ENode-facing ports must be untrusted ports.
- Network-facing (switch-facing) ports must be set in **trunk** interface mode.
- Network-facing ports must be FCoE trusted ports.
- Explicitly configure the beacon period. The beacon period is essentially a keepalive timer for virtual link maintenance.

When you enable VN2VF_Port FIP snooping, the system snoops VN_Port to VF_Port packets and enforces security only on VN_Port to VF_Port virtual links. When you enable VN2VN_Port FIP snooping, the system snoops VN_Port to VN_Port packets and enforces security only on VN_Port to VN_Port virtual links.

The transit switch applies VN2VN_Port FIP snooping filters at the ports associated with the FCoE VLANs on which you enable VN2VN FIP snooping.

This example describes how to configure VN2VN_Port FIP snooping when the FCoE hosts are directly connected to different transit switches, and the transit switches are directly connected to each other:

- **Requirements on page 153**
- **Overview on page 154**
- **Configuration on page 155**
- **Verification on page 157**

**Requirements**

This example uses the following hardware and software components:

- Two Juniper Networks QFX5100 Switches running the ELS CLI and used as transit switches
- Junos OS Release 13.2 or later for the QFX Series
- Two FCoE hosts that have ENodes

**Overview**

This example shows you how to:

- Set the correct interface mode on the transit switch.
- Configure the interfaces to use the dedicated FCoE VLAN for VN2VN_Port FIP snooping.
- Configure the network-facing interfaces as FCoE trusted interfaces.
- Configure the dedicated FCoE VLAN for VN2VN_PORT FIP snooping traffic.
- Enable VN2VN_PORT FIP snooping on the FCoE VLAN and configure the beacon period.

**Topology**

Table 17 on page 154 shows the configuration components for this example.

<table>
<thead>
<tr>
<th>Component</th>
<th>Settings</th>
</tr>
</thead>
</table>
| Hardware | Two QFX5100 switches running the ELS CLI (FCoE transit switch TS1 and FCoE transit switch TS2)  
  Two FCoE hosts that have ENodes (ENode1 and ENode2, respectively) |
| Interfaces and interface mode |  
  - Interface xe-0/0/20, interface mode trunk, connects directly from transit switch TS1 to the FCoE host with ENode1.  
  - Interface xe-0/0/21, interface mode trunk, connects directly from transit switch TS1 to transit switch TS2.  
  - Interface xe-0/0/31, interface mode trunk, connects directly from transit switch TS2 to transit switch TS1.  
  - Interface xe-0/0/30, interface mode trunk, connects directly from transit switch TS2 to the FCoE host with ENode2. |
| Interface VLAN membership | The interfaces on both transit switches use VLAN vlan200. |
| VN2VN_PORT FIP snooping VLAN | VLAN name (both transit switches)—vlan200  
  VLAN ID—200 |
| FIP snooping mode and beacon period | Set examine-vn2vn (VN2VN_PORT FIP snooping)  
  Beacon period—90000 ms |

Figure 9 on page 155 shows the network topology for this example.
Figure 9: VN2VN_Port FIP Snooping (FCoE Hosts Connected to Different Transit Switches) Topology

Configuration

To configure VN2VN_Port FIP snooping for VN_Ports that are directly connected to different transit switches (and the transit switches are directly connected to each other), perform these tasks:

- Configuring VN2VN_Port FIP Snooping on FCoE Transit Switch TS1 on page 156
- Configuring VN2VN_Port FIP Snooping on FCoE Transit Switch TS2 on page 156

The configuration for each FCoE transit switch is shown separately.

To quickly configure VN2VN_Port FIP snooping for FCoE hosts connected directly to different transit switches, 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. To configure FCoE transit switch TS1:

```
FCoE Transit Switch TS1

set interfaces xe-0/0/20 unit 0 family ethernet-switching interface-mode trunk
set interfaces xe-0/0/21 unit 0 family ethernet-switching interface-mode trunk
set interfaces xe-0/0/20 unit 0 family ethernet-switching vlan members vlan200
set interfaces xe-0/0/21 unit 0 family ethernet-switching vlan members vlan200
set vlans vlan200 vlan-id 200
set vlans vlan200 forwarding-options fip-security interface xe-0/0/21 fcoe-trusted
set vlans vlan200 forwarding-options fip-security examine-vn2vn beacon-period 90000
```

To quickly configure VN2VN_Port FIP snooping for FCoE hosts connected directly to different transit switches, 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. To configure FCoE transit switch TS2:

```
FCoE Transit Switch TS2

set interfaces xe-0/0/30 unit 0 family ethernet-switching interface-mode trunk
set interfaces xe-0/0/31 unit 0 family ethernet-switching interface-mode trunk
set interfaces xe-0/0/30 unit 0 family ethernet-switching vlan members vlan200
set interfaces xe-0/0/31 unit 0 family ethernet-switching vlan members vlan200
set vlans vlan200 vlan-id 200
set vlans vlan200 forwarding-options fip-security interface xe-0/0/31 fcoe-trusted
set vlans vlan200 forwarding-options fip-security examine-vn2vn beacon-period 90000
```
Configuring VN2VN_Port FIP Snooping on FCoE Transit Switch TS1

Step-by-Step Procedure
To configure interface mode, configure interface VLAN membership in the FCoE VLAN dedicated to VN2VN_Port traffic, set the network-facing port as FCoE trusted, configure the VLAN, set the beacon period, and enable VN2VN_Port FIP snooping:

1. Configure the modes of the interfaces that connect directly to the FCoE host with ENode1 (xe-0/0/20) and to FCoE transit switch TS2 (xe-0/0/21):

   user@switch# set interfaces xe-0/0/20 unit 0 family ethernet-switching interface-mode trunk
   set interfaces xe-0/0/21 unit 0 family ethernet-switching interface-mode trunk

2. Configure the interface VLAN membership so that the interfaces are members of the dedicated VN2VN_Port VLAN (vlan200):

   user@switch# set interfaces xe-0/0/20 unit 0 family ethernet-switching vlan members vlan200
   set interfaces xe-0/0/21 unit 0 family ethernet-switching vlan members vlan200

3. Configure the FCoE VLAN dedicated to VN2VN_Port FIP snooping:

   user@switch# set vlans vlan200 vlan-id 200

4. Configure the network-facing port (xe-0/0/21) as an FCoE trusted port:

   user@switch# set vlans vlan200 forwarding-options fip-security interface xe-0/0/21 fcoe-trusted

5. Enable VN2VN_Port FIP snooping on the VLAN and configure the beacon period:

   user@switch# set vlans vlan200 forwarding-options fip-security examine-vn2vn beacon-period 90000

Configuring VN2VN_Port FIP Snooping on FCoE Transit Switch TS2

Step-by-Step Procedure
To configure interface mode, configure interface VLAN membership in the FCoE VLAN dedicated to VN2VN_Port traffic, set the network-facing port as FCoE trusted, configure the VLAN, set the beacon period, and enable VN2VN_Port FIP snooping:

1. Configure the modes of the interfaces that connect directly to the FCoE host with ENode2 (xe-0/0/30) and to FCoE transit switch TS1 (xe-0/0/31):

   user@switch# set interfaces xe-0/0/30 unit 0 family ethernet-switching interface-mode trunk
   set interfaces xe-0/0/31 unit 0 family ethernet-switching interface-mode trunk

2. Configure the interface VLAN membership so that the interfaces are members of the dedicated VN2VN_Port VLAN (vlan200):

   user@switch# set interfaces xe-0/0/30 unit 0 family ethernet-switching vlan members vlan200
   set interfaces xe-0/0/31 unit 0 family ethernet-switching vlan members vlan200

3. Configure the FCoE VLAN dedicated to VN2VN_Port FIP snooping:

   user@switch# set vlans vlan200 vlan-id 200

4. Configure the network-facing port (xe-0/0/31) as an FCoE trusted port:
user@switch#  set vlans vlan200 forwarding-options fip-security interface xe-0/0/31 fcoe-trusted

5. Enable VN2VN_Port FIP snooping on the VLAN and configure the beacon period:
   
   user@switch#  set vlans vlan200 forwarding-options fip-security examine-vn2vn beacon-period 90000

Verification

To verify that the VN2VN_Port FIP snooping configuration has been created and is operating properly on both switches, perform these tasks:

- Verifying That VN2VN_Port FIP Snooping is Enabled on the FCoE VLAN (Transit Switches TS1 and TS2) on page 157

Verifying That VN2VN_Port FIP Snooping is Enabled on the FCoE VLAN (Transit Switches TS1 and TS2)

Purpose

Verify that VN2VN_Port FIP snooping is enabled on the correct VLAN (vlan200), the beacon period is set to 90000 milliseconds, and that the correct interfaces (xe-0/0/20 and xe-0/0/21 on TS1, and xe-0/0/30 and xe-0/0/31 on TS2) are members of the VLAN.
**Action**  
List the FIP snooping information on transit switch TS1 using the operational mode command `show fip snooping detail`

```bash
user@switch> show fip snooping detail
VLAN: vlan200, Mode: VN2VN Snooping
   FC-MAP: 0e:fc:00
   Beacon_Period: 90000
   VN2VN Mode: Point-to-Point
Enode Information
   Enode-MAC: 10:10:94:01:00:02,     Interface: xe-0/0/20
   Active VN_Ports : 1
   VN_Port Information
   VN-Port MAC: 0e:fc:00:01:0a:01
      Active Sessions : 1
      Session Information
        Vlink far-end VN-Port-MAC: 0e:fc:00:01:0b:01
Enode-MAC: 10:10:94:01:00:02,     Interface: xe-0/0/21
   Active VN_Ports : 1
   VN_Port Information
   VN-Port MAC: 0e:fc:00:01:0b:01
      Active Sessions : 1
      Session Information
        Vlink far-end VN-Port-MAC: 0e:fc:00:01:0a:01
```

List the FIP snooping information on transit switch TS2 using the operational mode command `show fip snooping detail`

```bash
user@switch> show fip snooping detail
VLAN: vlan200, Mode: VN2VN Snooping
   FC-MAP: 0e:fd:00
   Beacon_Period: 90000
   VN2VN Mode: Point-to-Point
Enode Information
   Enode-MAC: 10:10:94:01:00:02,     Interface: xe-0/0/30
   Active VN_Ports : 1
   VN_Port Information
   VN-Port MAC: 0e:fd:00:00:0b:01
      Active Sessions : 1
      Session Information
        Vlink far-end VN-Port-MAC: 0e:fd:00:00:0a:01
Enode-MAC: 10:10:94:01:00:02,     Interface: xe-0/0/31
   Active VN_Ports : 1
   VN_Port Information
   VN-Port MAC: 0e:fd:00:00:0a:01
      Active Sessions : 1
      Session Information
        Vlink far-end VN-Port-MAC: 0e:fd:00:00:0b:01
```

**Meaning**  
The `show fip snooping detail` command lists all of the transit switch information about VN2VN_Port FIP snooping and VN2VF_Port FIP snooping on each transit switch. The command shows that:

- The VLAN is `vlan200`.
- The mode is FIP snooping mode VN2VN, for VN2VN_Port FIP snooping. (If the Mode field shows VN2VF, then the FIP snooping mode is VN2VF_Port FIP snooping.)
The beacon period is **90000**.

The interfaces connected to the ENodes are `xe-0/0/20` and `xe-0/0/21` on transit switch TS1, and `xe-0/0/30` and `xe-0/0/31` on transit switch TS2. Because the transit switches are transparent passthrough switches, the network-facing trunk ports "see" the FCoE host ENodes at the far end of the VN2VN_Port virtual link.

In addition, this useful command shows information about the ENodes and the VN2VN_Port sessions.

---

**Related Documentation**

- Example: Configuring VN2VN_Port FIP Snooping (FCoE Hosts Directly Connected to the Same FCoE Transit Switch) on page 147
- Example: Configuring VN2VN_Port FIP Snooping (FCoE Hosts Indirectly Connected Through an Aggregation Layer FCoE Transit Switch) on page 159
- Enabling VN2VN_Port FIP Snooping and Configuring the Beacon Period on an FCoE Transit Switch on page 182
- Understanding VN_Port to VN_Port FIP Snooping on an FCoE Transit Switch on page 45

---

**Example: Configuring VN2VN_Port FIP Snooping (FCoE Hosts Indirectly Connected Through an Aggregation Layer FCoE Transit Switch)**

This example shows how to configure VN_Port to VN_Port (VN2VN_Port) FIP snooping when the hosts are indirectly connected through an aggregation layer FCoE transit switch. Each FCoE host ENode is directly connected to an FCoE transit switch, but the FCoE transit switches are not directly connected to each other. The FCoE transit switches are both connected to a third FCoE transit switch that acts as an aggregation layer switch.

---

**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 VN2VN_Port FIP Snooping (FCoE Hosts Indirectly Connected Through an Aggregation Layer FCoE Transit Switch). For ELS details, see Getting Started with Enhanced Layer 2 Software.

VN2VN_Port FIP snooping on an FCoE transit switch provides security to help prevent unauthorized access and data transmission on a bridge that connects ENodes in the Ethernet network. VN2VN_Port FIP snooping provides security for virtual links by creating filters based on information gathered (snooped) about FCoE devices during FIP transactions.

VN2VN_Port FIP snooping is conceptually similar to VN2VN_Port FIP snooping between VN_Ports and VF_Ports, but VN2VN_Port FIP snooping does not require traffic between VN_Ports to traverse the Fibre Channel (FC) switch or FCoE forwarder (FCF). Instead, a VN_Port communicates transparently through one or more transit switches on a virtual link that emulates a direct connection to the VN_Port at the other end of the virtual link.
To configure VN2VN_Port FIP snooping when the hosts are indirectly connected, you must follow these configuration rules:

- VN2VN_Port traffic must use a dedicated FCoE VLAN, and all ENodes that communicate using VN2VN_Port FIP snooping must use that FCoE VLAN. The FCoE VLAN must be configured on each transit switch. You cannot mix VN2VN_Port FIP snooping traffic with VN2VF_Port FIP snooping traffic in the same FCoE VLAN.

**NOTE:** An FCoE VLAN can support either VN2VF_Port FIP snooping or VN2VN_Port FIP snooping, but not both. Configure separate FCoE VLANs for VN2VF_Port FIP snooping traffic and for VN2VN_Port FIP snooping traffic. On FCoE VLANs that are configured as VN2VN_Port FIP snooping VLANs, VN_Port to VF_Port traffic is dropped.

- ENode-facing ports must be set in **trunk** interface mode.
- ENode-facing ports must be untrusted ports.
- Network-facing (switch-facing) ports must be set in **trunk** interface mode.
- Network-facing ports must be FCoE trusted ports.
- Explicitly configure the beacon period. The beacon period is essentially a keepalive timer for virtual link maintenance.

When you enable FIP snooping, the system snoops VN_Port to VF_Port packets and enforces security only on VN_Port to VF_Port virtual links. When you enable VN2VN_Port FIP snooping, the system snoops VN_Port to VN_Port packets and enforces security only on VN_Port to VN_Port virtual links.

The transit switch applies VN2VN_Port FIP snooping filters at the ports associated with the FCoE VLANs on which you enable VN2VN FIP snooping.

This example describes how to configure VN2VN_Port FIP snooping when the FCoE hosts are indirectly connected across an aggregation layer FCoE transit switch:

- **Requirements** on page 160
- **Overview** on page 161
- **Configuration** on page 162
- **Verification** on page 165

**Requirements**

This example uses the following hardware and software components:

- Three Juniper Networks QFX5100 Switches running the ELS CLI and used as transit switches
- Junos OS Release 13.2 or later for the QFX Series
- Two FCoE hosts that have ENodes
Overview

This example shows you how to:

- Set the correct interface mode on the transit switch.
- Configure the interfaces to use the dedicated FCoE VLAN for VN2VN_Port FIP snooping.
- Configure the network-facing interfaces as FCoE trusted interfaces.
- Configure the dedicated FCoE VLAN for VN2VN_Port FIP snooping traffic.
- Enable VN2VN_Port FIP snooping on the FCoE VLAN and configure the beacon period.

Topology

Table 18 on page 161 shows the configuration components for this example.

Table 18: Components of the VN2VN_Port FIP Snooping Configuration Topology (FCoE Hosts Indirectly Connected Across an Aggregation Layer FCoE Transit Switch)

<table>
<thead>
<tr>
<th>Component</th>
<th>Settings</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Hardware</strong></td>
<td>Three QFX5100 switches running the ELS CLI, two of which are FCoE transit switches that are directly attached to the FCoE hosts (transit switches TS1 and TS2) and one of which is an aggregation layer FCoE transit switch (TS3)</td>
</tr>
<tr>
<td></td>
<td>Two FCoE hosts that have ENodes (ENode1 and ENode2, respectively)</td>
</tr>
<tr>
<td>Interfaces and interface mode</td>
<td>Interface xe-0/0/20, interface mode <strong>trunk</strong>, connects directly from transit switch TS1 to the FCoE host with ENode1.</td>
</tr>
<tr>
<td></td>
<td>Interface xe-0/0/21, interface mode <strong>trunk</strong>, connects directly from transit switch TS1 to aggregation layer transit switch TS2.</td>
</tr>
<tr>
<td></td>
<td>Interface xe-0/0/31, interface mode <strong>trunk</strong>, connects directly from aggregation layer transit switch TS2 to transit switch TS1.</td>
</tr>
<tr>
<td></td>
<td>Interface xe-0/0/30, interface mode <strong>trunk</strong>, connects directly from aggregation layer transit switch TS2 to transit switch TS3.</td>
</tr>
<tr>
<td></td>
<td>Interface xe-0/0/11, interface mode <strong>trunk</strong>, connects directly from transit switch TS3 to aggregation layer transit switch TS2.</td>
</tr>
<tr>
<td></td>
<td>Interface xe-0/0/10, interface mode <strong>trunk</strong>, connects directly from transit switch TS3 to the FCoE host with ENode2.</td>
</tr>
<tr>
<td>Interface VLAN membership</td>
<td>The interfaces on all three switches use VLAN <strong>vlan200</strong>.</td>
</tr>
<tr>
<td>VN2VN_Port FIP snooping VLAN</td>
<td>VLAN name (all three switches)—<strong>vlan200</strong> VLAN ID—200</td>
</tr>
<tr>
<td>FIP snooping mode and beacon period</td>
<td><strong>Set examine-vn2vn</strong> (VN2VN_Port FIP snooping) Beacon period—90000 ms</td>
</tr>
</tbody>
</table>

Figure 10 on page 162 shows the network topology for this example.
Figure 10: VN2VN_Port FIP Snooping (FCoE Hosts Indirectly Connected)

Topology

Configuration

To configure VN2VN_Port FIP snooping for VN_Ports that are indirectly connected across an aggregation layer FCoE transit switch, perform these tasks:

- Configuring VN2VN_Port FIP Snooping on FCoE Transit Switch TS1 on page 163
- Configuring VN2VN_Port FIP Snooping on Aggregation Layer FCoE Transit Switch TS2 on page 164
- Configuring VN2VN_Port FIP Snooping on FCoE Transit Switch TS3 on page 164

CLI Quick Configuration

The configuration for each FCoE transit switch is shown separately.

To quickly configure VN2VN_Port FIP snooping for FCoE hosts that are indirectly connected across an aggregation layer FCoE transit switch, 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. To configure FCoE transit switch TS1:

```
set interfaces xe-0/0/20 unit 0 family ethernet-switching interface-mode trunk
set interfaces xe-0/0/21 unit 0 family ethernet-switching interface-mode trunk
set interfaces xe-0/0/20 unit 0 family ethernet-switching vlan members vlan200
set interfaces xe-0/0/21 unit 0 family ethernet-switching vlan members vlan200
set vlans vlan200 vlan-id 200
set vlans vlan200 forwarding-options fip-security interface xe-0/0/21 fcoe-trusted
set vlans vlan200 forwarding-options fip-security examine-vn2vn beacon-period 90000
```

To quickly configure VN2VN_Port FIP snooping for FCoE hosts that are indirectly connected across an aggregation layer FCoE transit switch, 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. To configure FCoE transit switch TS2:

```
set interfaces xe-0/0/30 unit 0 family ethernet-switching interface-mode trunk
set interfaces xe-0/0/31 unit 0 family ethernet-switching interface-mode trunk
set interfaces xe-0/0/30 unit 0 family ethernet-switching vlan members vlan200
set interfaces xe-0/0/31 unit 0 family ethernet-switching vlan members vlan200
set vlans vlan200 vlan-id 200
```
To quickly configure VN2VN_Port FIP snooping for FCoE hosts that are indirectly connected across an aggregation layer FCoE transit switch, 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. To configure FCoE transit switch TS3:

**FCoE Transit Switch**

**TS3**

```
set interfaces xe-0/0/10 unit 0 family ethernet-switching interface-mode trunk
set interfaces xe-0/0/11 unit 0 family ethernet-switching interface-mode trunk
set interfaces xe-0/0/10 unit 0 family ethernet-switching vlan members vlan200
set interfaces xe-0/0/11 unit 0 family ethernet-switching vlan members vlan200
set vlans vlan200 vlan-id 200
set vlans vlan200 forwarding-options fip-security interface xe-0/0/11 fcoe-trusted
set vlans vlan200 forwarding-options fip-security examine-vn2vn beacon-period 90000
```

**Configuring VN2VN_Port FIP Snooping on FCoE Transit Switch TS1**

**Step-by-Step Procedure**

To configure interface mode, configure interface VLAN membership in the FCoE VLAN dedicated to VN2VN_Port traffic, set the network-facing port as FCoE trusted, configure the VLAN, set the beacon period, and enable VN2VN_Port FIP snooping:

1. Configure the modes of the interfaces that connect directly to the FCoE host with ENode1 (xe-0/0/20) and to aggregation layer FCoE transit switch TS2 (xe-0/0/21):
   ```
   user@switch# set interfaces xe-0/0/20 unit 0 family ethernet-switching interface-mode trunk
   set interfaces xe-0/0/21 unit 0 family ethernet-switching interface-mode trunk
   ```

2. Configure the interface VLAN membership so that the interfaces are members of the dedicated VN2VN_Port VLAN (vlan200):
   ```
   user@switch# set interfaces xe-0/0/20 unit 0 family ethernet-switching vlan members vlan200
   set interfaces xe-0/0/21 unit 0 family ethernet-switching vlan members vlan200
   ```

3. Configure the FCoE VLAN dedicated to VN2VN_Port FIP snooping:
   ```
   user@switch# set vlans vlan200 vlan-id 200
   ```

4. Configure the network-facing port (xe-0/0/21) as an FCoE trusted port:
   ```
   user@switch# set vlans vlan200 forwarding-options fip-security interface xe-0/0/21 fcoe-trusted
   ```

5. Enable VN2VN_Port FIP snooping on the VLAN and configure the beacon period:
   ```
   user@switch# set vlans vlan200 forwarding-options fip-security examine-vn2vn beacon-period 90000
   ```
Step-by-Step Procedure

**Configuring VN2VN_Port FIP Snooping on Aggregation Layer FCoE Transit Switch TS2**

To configure interface mode, configure interface VLAN membership in the FCoE VLAN dedicated to VN2VN_Port traffic, set the network-facing ports as FCoE trusted, configure the VLAN, set the beacon period, and enable VN2VN_Port FIP snooping:

1. Configure the mode of the interfaces that connect directly to FCoE transit switches TS1 (xe-0/0/31) and TS3 (xe-0/0/30). Both interfaces are network-facing and must be configured as trunk interfaces:
   ```
   user@switch# set interfaces xe-0/0/30 unit 0 family ethernet-switching interface-mode trunk
   set interfaces xe-0/0/31 unit 0 family ethernet-switching interface-mode trunk
   ```
2. Configure the interface VLAN membership so that the interfaces are members of the dedicated VN2VN_Port VLAN (vlan200):
   ```
   user@switch# set interfaces xe-0/0/30 unit 0 family ethernet-switching vlan members vlan200
   set interfaces xe-0/0/31 unit 0 family ethernet-switching vlan members vlan200
   ```
3. Configure the FCoE VLAN dedicated to VN2VN_Port FIP snooping:
   ```
   user@switch# set vlans vlan200 vlan-id 200
   ```
4. Configure the network-facing ports (xe-0/0/30 and xe-0/0/31) as FCoE trusted ports:
   ```
   user@switch# set vlans vlan200 forwarding-options fip-security interface xe-0/0/30 fcoe-trusted
   user@switch# set vlans vlan200 forwarding-options fip-security interface xe-0/0/31 fcoe-trusted
   ```
5. Enable VN2VN_Port FIP snooping on the VLAN and configure the beacon period:
   ```
   user@switch# set vlans vlan200 forwarding-options fip-security examine-vn2vn beacon-period 90000
   ```

**Configuring VN2VN_Port FIP Snooping on FCoE Transit Switch TS3**

To configure interface mode, configure interface VLAN membership in the FCoE VLAN dedicated to VN2VN_Port traffic, set the network-facing port as FCoE trusted, configure the VLAN, set the beacon period, and enable VN2VN_Port FIP snooping:

1. Configure the mode of the interfaces that connect directly to the FCoE host with ENode2 (xe-0/0/10) and to aggregation layer FCoE transit switch TS2 (xe-0/0/11):
   ```
   user@switch# set interfaces xe-0/0/10 unit 0 family ethernet-switching interface-mode trunk
   set interfaces xe-0/0/11 unit 0 family ethernet-switching interface-mode trunk
   ```
2. Configure the interface VLAN membership so that the interfaces are members of the dedicated VN2VN_Port VLAN (vlan200):
   ```
   user@switch# set interfaces xe-0/0/10 unit 0 family ethernet-switching vlan members vlan200
   set interfaces xe-0/0/11 unit 0 family ethernet-switching vlan members vlan200
   ```
3. Configure the FCoE VLAN dedicated to VN2VN_Port FIP snooping:
user@switch# set vlans vlan200 vlan-id 200

4. Configure the network-facing port (xe-0/0/11) as an FCoE trusted port:
   user@switch# set vlans vlan200 forwarding-options fip-security interface xe-0/0/11 fcoe-trusted

5. Enable VN2VN_Port FIP snooping on the VLAN and configure the beacon period:
   user@switch# set vlans vlan200 forwarding-options fip-security examine-vn2vn beacon-period 90000

Verification

To verify that the VN2VN_Port FIP snooping configuration has been created and is operating properly on all three switches, perform these tasks:

- Verifying That VN2VN_Port FIP Snooping Is Enabled on the FCoE VLAN (All Three Transit Switches) on page 165

Verifying That VN2VN_Port FIP Snooping Is Enabled on the FCoE VLAN (All Three Transit Switches)

Purpose

Verify that VN2VN_Port FIP snooping is enabled on the correct VLAN (vlan200), the beacon period is set to 90000 milliseconds, and that the correct interfaces (xe-0/0/20 and xe-0/0/21 on TS1, xe-0/0/30 and xe-0/0/31 aggregation layer TS2, and xe-0/0/10 and xe-0/0/11 on TS3) are members of the VLAN.
**Action**

List the FIP snooping information on transit switch TS1 using the operational mode command `show fip snooping detail`

```
user@switch> show fip snooping detail
VLAN: vlan200, Mode: VN2VN Snooping
FC-MAP: 0e:fc:00
Beacon_Period: 90000
VN2VN Mode: Point-to-Point
Enode Information
Enode-MAC: 10:10:94:01:00:02, Interface: xe-0/0/20
  Active VN_Ports : 1
  VN_Port Information
  VN-Port MAC: 0e:fc:00:01:0a:01
  Active Sessions : 1
  Session Information
Vlink far-end VN-Port-MAC: 0e:fc:00:01:0b:01
Enode-MAC: 10:10:94:01:00:02, Interface: xe-0/0/21
  Active VN_Ports : 1
  VN_Port Information
  VN-Port MAC: 0e:fc:00:01:0b:01
  Active Sessions : 1
  Session Information
Vlink far-end VN-Port-MAC: 0e:fc:00:01:0a:01
```

List the FIP snooping information on aggregation layer transit switch TS2 using the operational mode command `show fip snooping detail`

```
user@switch> show fip snooping detail
VLAN: vlan200, Mode: VN2VN Snooping
FC-MAP: 0e:fc:00
Beacon_Period: 90000
VN2VN Mode: Point-to-Point
Enode Information
Enode-MAC: 10:10:94:01:00:02, Interface: xe-0/0/30
  Active VN_Ports : 1
  VN_Port Information
  VN-Port MAC: 0e:fc:00:01:0b:01
  Active Sessions : 1
  Session Information
Vlink far-end VN-Port-MAC: 0e:fc:00:01:0a:01
Enode-MAC: 10:10:94:01:00:02, Interface: xe-0/0/31
  Active VN_Ports : 1
  VN_Port Information
  VN-Port MAC: 0e:fc:00:01:0a:01
  Active Sessions : 1
  Session Information
Vlink far-end VN-Port-MAC: 0e:fc:00:01:0b:01
```

List the FIP snooping information on transit switch TS3 using the operational mode command `show fip snooping detail`

```
user@switch> show fip snooping detail
VLAN: vlan200, Mode: VN2VN Snooping
FC-MAP: 0e:fd:00
Beacon_Period: 90000
VN2VN Mode: Point-to-Point
Enode Information
Enode-MAC: 10:10:94:01:00:02, Interface: xe-0/0/40
  Active VN_Ports : 1
  VN_Port Information
  VN-Port MAC: 0e:fd:00:01:0b:01
  Active Sessions : 1
  Session Information
Vlink far-end VN-Port-MAC: 0e:fd:00:01:0a:01
```

Copyright © 2014, Juniper Networks, Inc.
Meaning  The `show fip snooping detail` command lists all of the transit switch information about VN2VN_Port FIP snooping and VN2VF_Port FIP snooping on each transit switch. The command shows that:

- The VLAN is `vlan200`.
- The mode is FIP snooping mode `VN2VN`, for VN2VN_Port FIP snooping. (If the Mode field shows `VN2VF`, then the FIP snooping mode is VN2VF_Port FIP snooping.)
- The beacon period is `90000`.
- The interfaces connected to the ENodes are `xe-0/0/20` and `xe-0/0/21` on transit switch TS1, `xe-0/0/30` and `xe-0/0/31` on aggregation layer transit switch TS2, and `xe-0/0/10` and `xe-0/0/11` on transit switch TS3. Because the transit switches are transparent passthrough switches, the network-facing trunk ports “see” the FCoE host ENodes at the far end of the VN2VN_Port virtual link.

In addition, this useful command shows information about the ENodes and the VN2VN_Port sessions.

Related Documentation  
- Example: Configuring VN2VN_Port FIP Snooping (FCoE Hosts Directly Connected to the Same FCoE Transit Switch) on page 147
- Example: Configuring VN2VN_Port FIP Snooping (FCoE Hosts Directly Connected to Different FCoE Transit Switches) on page 152
- Enabling VN2VN_Port FIP Snooping and Configuring the Beacon Period on an FCoE Transit Switch on page 182
- Understanding VN_Port to VN_Port FIP Snooping on an FCoE Transit Switch on page 45
CHAPTER 6

Configuration Tasks (Fibre Channel, FCoE, FIP, and FIP Snooping)

- Configuring a Physical Fibre Channel Interface on page 170
- Enabling and Disabling CoS OxID Hash Control on Standalone Switches on page 171
- Setting the Maximum Number of FIP Login Sessions per ENode on page 172
- Setting the Maximum Number of FIP Login Sessions per Node Device on page 173
- Configuring VLANs for FCoE Traffic on an FCoE Transit Switch on page 174
- Configuring VN2VF_Port FIP Snooping and FCoE Trusted Interfaces on an FCoE Transit Switch on page 179
- Enabling VN2VN_Port FIP Snooping and Configuring the Beacon Period on an FCoE Transit Switch on page 182
Configuring a Physical Fibre Channel Interface

When you configure the switch as an FCoE-FC gateway, you must configure either 6 or 12 of the physical interfaces as native FC interfaces. Native FC interfaces connect to the storage area network (SAN) FC switch.

You can configure ports xe-0/0/0 through xe-0/0/5 as fc-0/0/0 through fc-0/0/5, and ports xe-0/0/42 through xe-0/0/47 as fc-0/0/42 through fc-0/0/47 to create blocks of native FC interfaces. You cannot individually configure a single port as a native FC interface. Within these port blocks, you cannot mix FC interfaces with Ethernet interfaces. All of the ports in a block must be either native FC interfaces or Ethernet interfaces.

You can configure:

- Six native FC interfaces by configuring either ports xe-0/0/0 through xe-0/0/5 as fc-0/0/0 through fc-0/0/5, or ports xe-0/0/42 through xe-0/0/47 as fc-0/0/42 through fc-0/0/47.
- Twelve native FC interfaces by configuring ports xe-0/0/0 through xe-0/0/5 as fc-0/0/0 through fc-0/0/5 and ports xe-0/0/42 through xe-0/0/47 as fc-0/0/42 through fc-0/0/47.
- No native FC interfaces by leaving ports xe-0/0/0 through xe-0/0/5 and ports xe-0/0/42 through xe-0/0/47 in their default state as Ethernet interfaces.

To configure physical FC interfaces using the CLI, specify the physical port block you want to configure on the switch as native FC interfaces:

```
[edit chassis]
user@switch# set fpc fpc pic fibre-channel port-range port-range-low port-range-high
```

For example, to configure six native FC interfaces, you can configure ports 0 through 5 as physical FC interfaces:

```
[edit chassis]
user@switch# set fpc 0 pic 0 fibre-channel port-range 0 5
```

To configure 12 native FC interfaces requires two separate statements:

```
[edit chassis]
user@switch# set fpc 0 pic 0 fibre-channel port-range 0 5
user@switch# set fpc 0 pic 0 fibre-channel port-range 42 47
```

Related Documentation

- Configuring an FCoE-FC Gateway Fibre Channel Fabric
- Configuring an FCoE VLAN Interface on an FCoE-FC Gateway
- Configuring a Fibre Channel Interface
- Assigning Interfaces to a Fibre Channel Fabric
- Example: Setting Up Fibre Channel and FCoE VLAN Interfaces in an FCoE-FC Gateway Fabric
- Understanding Interfaces on an FCoE-FC Gateway
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 37
- Understanding OxID Hash Control for FCoE Traffic Load Balancing on QFabric Systems
- Understanding CoS IEEE 802.1p Priorities for Lossless Traffic Flows
Setting the Maximum Number of FIP Login Sessions per ENode

When the switch acts as an FCoE-FC gateway, FCoE node (ENode) devices in the Ethernet network use the gateway to connect to the Fibre Channel (FC) storage area network (SAN). You can limit the maximum number of FIP login sessions permitted on each ENode. Limiting the number of login sessions can prevent login session rejections caused when the connected FC switch port configuration limits the number of FIP login sessions.

The maximum number of FIP sessions per ENode is 2000 sessions (FLOGI plus FDISC sessions). The limit you set applies to every ENode in the specified gateway fabric. Each ENode in the fabric can have up to the maximum number of sessions, but the total number of active sessions cannot exceed the session limits you apply to the fabric or the Node device.

There are also configurable FIP login session limits that you can apply to the gateway FC fabric, to the QFX3500 switch or QFabric system Node device, and to the interfaces in each FC fabric.

- To set a maximum number of FIP login sessions per ENode using the CLI:
  
  ```
  [edit fc-fabrics fc-fabric-name protocols fip]
  user@switch# set max-sessions-per-enode max-login-sessions
  ```

  For example, to configure the ENodes on an FC fabric named `sanfab1` with a maximum FIP login session limit of 250 sessions:

  ```
  [edit fc-fabrics sanfab1]
  user@switch# set protocols fip max-sessions-per-enode 250
  ```

Related Documentation

- Setting the Maximum Number of FIP Login Sessions per FC Interface
- Setting the Maximum Number of FIP Login Sessions per FC Fabric
- Setting the Maximum Number of FIP Login Sessions per Node Device on page 173
- Understanding Interfaces on an FCoE-FC Gateway
Setting the Maximum Number of FIP Login Sessions per Node Device

When a QFX3500 switch or QFabric system Node device acts an FCoE-FC gateway, it connects FCoE devices on an Ethernet network to an FC switch in a Fibre Channel network. You can limit the maximum number of FIP login sessions for the FCoE devices on each Node device.

For QFX3500 switches, the maximum limit means that the sum of the FIP login sessions on all of the local FC fabrics on that QFX3500 switch cannot exceed the device maximum.

For the QFabric system, the limit applies to each Node device in the QFabric system. For example, if you configure a maximum FIP login session value of 2000 sessions, each Node device in the QFabric system can have a total of up to 2000 FIP login sessions running on its FC fabrics.

The maximum number of FIP sessions a device can support is 2500 sessions. (This is the combined total of all VN2VF_Port and VN2VN_Port sessions on the system.)

There are also configurable FIP login session limits that you can apply to the FC fabrics on the devices, to the NP_Port interfaces in each FC fabric, and to the ENodes in each FC fabric. To prevent unexpected FIP login rejections:

- The sum of the maximum FIP login sessions for all of the FC fabrics on a device should not exceed the maximum number of sessions per device.
- The sum of the maximum FIP login sessions on all of the NP_Port interfaces that belong to an FC fabric should not exceed the maximum number of sessions the FC fabric supports or the device supports.

- To set a maximum number of FIP login sessions for Node devices using the CLI:

  ```
  [edit fc-options]
  user@switch# set max-login-sessions-per-node max-login-sessions-per-node
  ```

  For example, to configure a maximum FIP login limit of 2000 sessions on a QFX3500 switch or on all Node devices in a QFabric system:

  ```
  [edit fc-options]
  user@switch# set max-login-sessions-per-node 2000
  ```

Related Documentation
- Setting the Maximum Number of FIP Login Sessions per ENode on page 172
- Setting the Maximum Number of FIP Login Sessions per FC Interface
- Setting the Maximum Number of FIP Login Sessions per FC Fabric
- Understanding Interfaces on an FCoE-FC Gateway
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.
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
   ```
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 ethernet-switching interface-mode trunk 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 ethernet-switching interface-mode trunk 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 20
- Understanding FCoE Transit Switch Functionality on page 26
- Example: Configuring CoS PFC for FCoE Traffic on page 115
- Configuring VN2VF_Port FIP Snooping and FCoE Trusted Interfaces on an FCoE Transit Switch on page 179
- Enabling VN2VN_Port FIP Snooping and Configuring the Beacon Period on an FCoE Transit Switch on page 182
- Disabling Enhanced FIP Snooping Scaling
- Configuring an FCoE LAG
- Configuring an FCoE VLAN Interface on an FCoE-FC Gateway

**Configuring VN2VF_Port FIP Snooping and FCoE Trusted Interfaces on an FCoE Transit Switch**

VN_Port to VF_Port (VN2VF_Port) Fibre Channel over Ethernet (FCoE) Initialization Protocol (FIP) snooping uses information gathered during FIP discovery and login to create firewall filters that provide security against unauthorized access to the FC switch or FCoE forwarder (FCF) through the switch when the switch is acting as an FCoE transit switch. The firewall filters allow only FCoE devices that successfully log in to the FC fabric to access the FCF through the transit switch. VN2VF_Port FIP snooping provides security for the point-to-point virtual links that connect host FCoE Nodes (ENodes) and FCFs in the FCoE VLAN by denying access to any device that does not successfully log in to the FCF.

VN2VF_Port FIP snooping is disabled by default. You enable VN2VF_Port FIP snooping on a per-VLAN basis for VLANs that carry FCoE traffic. Ensure that a VLAN that carries FCoE traffic carries only FCoE traffic, because enabling VN2VF_Port FIP snooping denies access for all other Ethernet traffic.
NOTE: All of the transit switch ports are untrusted by default. If an ENode on an FCoE device logs in to an FCF before you enable VN2VF_Port FIP snooping on the VLAN and you then enable VN2VF_Port FIP snooping, the transit switch denies traffic from the ENode because the transit switch has not snooped (learned) the ENode state. The following process automatically logs the ENode back in to the FCF to reestablish the connection:

1. VN2VF_Port FIP snooping is enabled on an FCoE VLAN on the switch.
2. The switch denies existing connections between servers and the FCF on the FCoE VLAN by filtering the FCoE traffic and FIP traffic, so no keepalive messages from the ENodes reach the FCF.
3. The FCF port timer for each ENode and for each VN_Port on each ENode expires.
4. The FCF sends each ENode whose port timer has expired a Clear Virtual Links (CVL) message.
5. The CVL message causes the ENode to log in again.

Because the FCF is a trusted source, you configure interfaces that connect to the FCF as FCoE trusted interfaces. FCoE trusted interfaces do not filter traffic (FIP snooping filtering should occur only at the FCoE access edge), but VN2VF_Port FIP snooping continues to run on trusted interfaces so that the switch learns the FCF state.

NOTE: Do not configure ENode-facing interfaces both with FIP snooping enabled and as trusted interfaces. FCoE VLANs with interfaces that are directly connected to FCoE hosts should be configured with FIP snooping enabled and the interfaces should not be trusted interfaces. Ethernet interfaces that are connected to an FCF should be configured as trusted interfaces and should not have FIP snooping enabled. Interfaces that are connected to a transit switch that is performing FIP snooping can be configured as trusted interfaces if the FCoE VLAN is not enabled for FIP snooping.

Optionally, you can specify an FC-MAP value for each FCoE VLAN. On a given FCoE VLAN, the switch learns only FCFs that have a matching FC-MAP value. The default FC-MAP value is 0EFC00h for all FC devices. (Enter hexadecimal values for FC-MAP preceded by the hexadecimal indicator "0x"—for example, 0x0EFC00.) If you change the FC-MAP value of an FCF, change the FC-MAP value for the FCoE VLAN it belongs to on the switch and on the servers you want to communicate with the FCF. An FCoE VLAN can have one and only one FC-MAP value.
NOTE: The default enhanced FIP snooping scaling supports 2,500 sessions. On QFabric systems, starting with Junos OS Release 13.2X52, you can disable enhanced FIP snooping scaling on a per-VLAN basis if you want to do so, but only 376 sessions are supported if you disable enhanced FIP snooping scaling.

There are differences in the way you configure FIP snooping and FCoE trusted interfaces on a switch that depend on whether the switch uses the original CLI or the Enhanced Layer 2 Software (ELS) CLI. 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 enable VN2VF_Port FIP snooping:

- To enable VN2VF_Port FIP snooping on a single VLAN and specify the optional FC-MAP value:

  ```
  [edit ethernet-switching-options secure-access-port]
  user@switch# set vlan vlan-name examine-fip fc-map fc-map-value
  ```

  For example, to enable VN2VF_Port FIP snooping on a VLAN named san1_vlan and change the FC-MAP value to 0x0EFC03:

  ```
  [edit ethernet-switching-options secure-access-port]
  user@switch# set vlan san1_vlan examine-fip fc-map 0x0EFC03
  ```

  **NOTE:** Changing the FC-MAP value causes all logins to drop and forces ENodes to log in again.

- To enable VN2VF_Port FIP snooping on all VLANs and use the default FC-MAP value:

  ```
  [edit ethernet-switching-options secure-access-port]
  user@switch# set vlan all examine-fip
  ```

- To configure an interface as an FCoE trusted interface:

  ```
  [edit ethernet-switching-options secure-access-port]
  user@switch# set interface interface-name fcoe-trusted
  ```

  For example, to configure interface xe-0/0/30 as an FCoE trusted interface:

  ```
  [edit ethernet-switching-options secure-access-port]
  user@switch# set interface xe-0/0/30 fcoe-trusted
  ```

### ELS CLI Configuration

To enable VN2VF_Port FIP snooping:

- To enable VN2VF_Port FIP snooping on a VLAN and specify the optional FC-MAP value:

  ```
  [edit]
  user@switch# set vlans vlan-name forwarding-options fip-security fc-map fc-map-value examine-vn2vf
  ```

  For example, to enable VN2VF_Port FIP snooping on a VLAN named san1_vlan and change the FC-MAP value to 0x0EFC03:
[edit]
user@switch# set vlans san1_vlan forwarding-options fip-security fc-map 0x0EFC03
examine-vn2vf

**NOTE:** Changing the FC-MAP value causes all logins to drop and forces ENodes to log in again.

- To configure an interface as an FCoE trusted interface:

  [edit]
  user@switch# set vlans vlan-name forwarding-options fip-security interface interface-name fcoe-trusted

  For example, to configure interface xe-0/0/30 on VLAN named san1_vlan as an FCoE trusted interface:

  [edit]
  user@switch# set vlans san1_vlan forwarding-options fip-security interface xe-0/0/30 fcoe-trusted

**Related Documentation**

- Example: Configuring an FCoE Transit Switch
- Configuring an FCoE VLAN Interface on an FCoE-FC Gateway
- Configuring VLANs for FCoE Traffic on an FCoE Transit Switch on page 174
- Configuring an FCoE LAG
- Disabling Enhanced FIP Snooping Scaling
- Understanding FIP Snooping
- Understanding VN_Port to VF_Port FIP Snooping on an FCoE Transit Switch on page 38
- Understanding FCoE LAGs

**Enabling VN2VN_Port FIP Snooping and Configuring the Beacon Period on an FCoE Transit Switch**

VN_Port to VN_Port (VN2VN_Port) FIP snooping on an FCoE transit switch provides security to help prevent unauthorized access and data transmission on a bridge that connects ENodes in the Ethernet network. VN2VN_Port FIP snooping provides security for virtual links by creating filters based on information gathered (snooped) about FCoE devices during FIP transactions.

VN2VN_Port FIP snooping is conceptually similar to VN2VF_Port FIP snooping between VN_Ports and VF_Ports, but VN2VN_Port FIP snooping does not require traffic between VN_Ports to traverse the Fibre Channel (FC) switch or FCoE forwarder (FCF). Instead, a VN_Port communicates transparently through the transit switch on a virtual link that emulates a direct connection to the VN_Port at the other end of the virtual link.

VN2VN_Port FIP snooping is disabled by default. You enable VN2VN_Port FIP snooping on a per-VLAN basis on VLANs that carry VN2VN_Port FCoE traffic. Ensure that the VLAN
carries only FCoE traffic between VN_Ports, because enabling VN2VN_Port FIP snooping denies access for all other traffic, including VN2VF_Port FIP snooping traffic.

All ENodes that you want to communicate using VN2VN_Port FIP snooping must use an FCoE VLAN dedicated to VN2VN_Port traffic. You cannot mix VN2VN_Port FIP snooping traffic with VN2VF_Port FIP snooping traffic in the same FCoE VLAN.

NOTE: An FCoE VLAN can support either VN2VF_Port FIP snooping or VN2VN_Port FIP snooping, but not both. Configure separate FCoE VLANs for VN2VF_Port FIP snooping traffic and for VN2VN_Port FIP snooping traffic. On FCoE VLANs that are configured as VN2VN_Port FIP snooping VLANs, VN2VF_Port traffic is dropped.

The beacon period is conceptually similar to the FIP keepalive period (timer) for VN2VF_Port FIP snooping virtual link maintenance. The beacon period performs virtual link maintenance for VN2VN_Port FIP snooping. It is the time interval between messages that verify the connection is still valid and the device at the other end of the virtual link is still reachable. You set the beacon period value for each FCoE VLAN that you configure to do VN2VN_Port FIP snooping.

NOTE: In addition to enabling VN2VN_Port FIP snooping and configuring the beacon period, you must also configure a dedicated FCoE VLAN for the VN2VN_Port traffic, and set the FCoE transit switch ports in the proper port mode and trusted or untrusted state (interfaces are untrusted by default). See the VN2VN_Port FIP snooping configuration example topics for complete configurations of several common network topologies.

There are differences in the way you configure a native VLAN on an interface that depend on whether the switch uses the original CLI or the Enhanced Layer 2 Software (ELS) CLI. 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 enable VN2VN_Port FIP snooping and set the beacon period on an FCoE VLAN that is dedicated to VN2VN_Port traffic:

- [edit ethernet-switching-options secure-access-port]

```plaintext
user@switch# set vlan vlan-name examine-fip examine-vn2vn beacon-period milliseconds
```

For example, to enable VN2VN_Port FIP snooping on a VLAN named `vlan200` and set the beacon period to `90000` milliseconds:

```plaintext
[edit ethernet-switching-options secure-access-port]
user@switch# set vlan vlan200 examine-fip examine-vn2vn beacon-period 90000
```

**ELS CLI Configuration**
To enable VN2VN_Port FIP snooping and set the beacon period on an FCoE VLAN that is dedicated to VN2VN_Port traffic:

- [edit]
  user@switch# set vlans vlan-name forwarding-options fip-security examine-vn2vn beacon-period milliseconds

For example, to enable VN2VN_Port FIP snooping on a VLAN named `vlan200` and set the beacon period to `90000` milliseconds:

- [edit]
  user@switch# set vlans vlan200 forwarding-options fip-security examine-vn2vn beacon-period 90000

**Related Documentation**

- Example: Configuring VN2VN_Port FIP Snooping (FCoE Hosts Directly Connected to the Same FCoE Transit Switch)
- Example: Configuring VN2VN_Port FIP Snooping (FCoE Hosts Directly Connected to Different FCoE Transit Switches)
- Example: Configuring VN2VN_Port FIP Snooping (FCoE Hosts Indirectly Connected Through an Aggregation Layer FCoE Transit Switch)
- Configuring VN2VF_Port FIP Snooping and FCoE Trusted Interfaces on an FCoE Transit Switch on page 179
- Understanding VN_Port to VN_Port FIP Snooping on an FCoE Transit Switch on page 45
CHAPTER 7
Configuration Tasks (DCBX)

- Configuring the DCBX Mode on page 185
- Configuring DCBX Autonegotiation on page 186
- Disabling the ETS Recommendation TLV on page 189
- Defining an Application for DCBX Application Protocol TLV Exchange on page 189
- Configuring an Application Map for DCBX Application Protocol TLV Exchange on page 191
- Applying an Application Map to an Interface for DCBX Application Protocol TLV Exchange on page 192

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 186
- Disabling the ETS Recommendation TLV on page 189
- Understanding DCBX on page 87
- Understanding DCBX Application Protocol TLV Exchange on page 97
- show dcbx neighbors on page 249

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 enhanced transmission selection (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-FC gateway, it does not send or receive FCoE Initialization Protocol (FIP) packets.
- 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.”

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 105
- Example: Configuring CoS PFC for FCoE Traffic on page 115
- Disabling the ETS Recommendation TLV on page 189
- Understanding DCBX Application Protocol TLV Exchange on page 97
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 185
- Configuring DCBX Autonegotiation on page 186
- Understanding DCBX on page 87
- *Understanding Data Center Bridging Capability Exchange Protocol for EX Series Switches*

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 191
- Applying an Application Map to an Interface for DCBX Application Protocol TLV Exchange on page 192
- Configuring DCBX Autonegotiation on page 186
- Example: Configuring DCBX Application Protocol TLV Exchange on page 105
- Example: Configuring DCBX to Support an iSCSI Application
- Understanding DCBX Application Protocol TLV Exchange on page 97
- `show dcbx neighbors` on page 249
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-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-name ptp code points [ 001 101 ]

Related Documentation

- Defining an Application for DCBX Application Protocol TLV Exchange on page 189
- Applying an Application Map to an Interface for DCBX Application Protocol TLV Exchange on page 192
- Configuring DCBX Autonegotiation on page 186
- Example: Configuring DCBX Application Protocol TLV Exchange on page 105
- Example: Configuring DCBX to Support an iSCSI Application
- show dcbx neighbors on page 249
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 189
- Configuring an Application Map for DCBX Application Protocol TLV Exchange on page 191
- Configuring DCBX Autonegotiation on page 186
- Example: Configuring DCBX Application Protocol TLV Exchange on page 105
- Example: Configuring DCBX to Support an iSCSI Application
- show dcbx neighbors on page 249
CHAPTER 8

Configuration Statements

• application (Application Maps) on page 194
• application (Applications) on page 195
• application-map on page 196
• application-maps on page 197
• applications (Applications) on page 198
• applications (DCBX) on page 199
• beacon-period on page 200
• code-points (Application Maps) on page 201
• dcbx on page 202
• dcbx-version on page 203
• destination-port (Applications) on page 204
• disable (DCBX) on page 205
• enhanced-transmission-selection on page 206
• ether-type on page 207
• examine-vn2vn on page 208
• fc-map on page 209
• fcoe-trusted on page 211
• interface (DCBX) on page 212
• no-recommendation-tlv on page 213
• policy-options on page 214
• priority-flow-control on page 215
• protocol (Applications) on page 216
• recommendation-tlv on page 217
application (Application Maps)

Syntax

```plaintext
application application-name {
    code-points [ aliases ] [ bit-patterns ];
}
```

Hierarchy Level

[edit policy-options application-maps application-map-name]

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

- `application-name`—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

- Configuring an Application Map for DCBX Application Protocol TLV Exchange on page 191
- Example: Configuring DCBX Application Protocol TLV Exchange on page 105
- Example: Configuring DCBX to Support an iSCSI Application
- Understanding DCBX Application Protocol TLV Exchange on page 97
- Understanding DCBX Application Protocol TLV Exchange on EX Series Switches
application (Applications)

Syntax

```plaintext
application application-name {
  destination-port port-value;
  protocol (tcp | udp);
  ether-type type;
}
```

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

- `application-name`—Name of the application.

The statements are explained separately.

Required Privilege

- interface—to view this statement in the configuration.
- interface-control—to add this statement to the configuration.

Related Documentation

- Defining an Application for DCBX Application Protocol TLV Exchange on page 189
- Example: Configuring DCBX Application Protocol TLV Exchange on page 105
- Example: Configuring DCBX to Support an iSCSI Application
- Understanding DCBX Application Protocol TLV Exchange on page 97
- Understanding DCBX Application Protocol TLV Exchange on EX Series Switches
application-map

Syntax    application-map application-map-name;

Hierarchy Level    [edit protocols dcbx interface interface-name]

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    application-map-name—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    • show dcbx neighbors on page 249
    • Applying an Application Map to an Interface for DCBX Application Protocol TLV Exchange on page 192
    • Example: Configuring DCBX Application Protocol TLV Exchange on page 105
    • Example: Configuring DCBX to Support an iSCSI Application
    • Understanding DCBX Application Protocol TLV Exchange on page 97
    • Understanding DCBX Application Protocol TLV Exchange on EX Series Switches
**application-maps**

**Syntax**

```
application-maps application-map-name {
  application application-name {
    code-points [ aliases ] [ bit-patterns ];
  }
}
```

**Hierarchy Level**

[edit policy-options]

**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 an application map by specifying the applications that belong to the application map.

**Options**

- `application-map-name`—Name of the application map.
  
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**

- Configuring an Application Map for DCBX Application Protocol TLV Exchange on page 191
- Example: Configuring DCBX Application Protocol TLV Exchange on page 105
- Example: Configuring DCBX to Support an iSCSI Application
- Understanding DCBX Application Protocol TLV Exchange on page 97
- Understanding DCBX Application Protocol TLV Exchange on EX Series Switches
applications (Applications)

Syntax

```
applications {
  application application-name {
    destination-port port-value;
    protocol (tcp | udp);
    ether-type type;
  }
}
```

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
- Defining an Application for DCBX Application Protocol TLV Exchange on page 189
- Example: Configuring DCBX Application Protocol TLV Exchange on page 105
- Example: Configuring DCBX to Support an iSCSI Application
- Understanding DCBX Application Protocol TLV Exchange on page 97
- Understanding DCBX Application Protocol TLV Exchange on EX Series Switches
## applications (DCBX)

**Syntax**

```plaintext
applications {
    no-auto-negotiation;
}
```

**Hierarchy Level**

```
[edit protocols dcbx interface interface-name]
```

**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**
- show dcbx neighbors on page 249
- Understanding DCB Features and Requirements on page 16
beacon-period

**Syntax**

`beacon-period milliseconds;`

**Hierarchy Level**

Original CLI

```
[edit ethernet-switching options secure-access-port vlan (all | vlan-name) examine-fip examine-vn2vn]
```

ELS CLI for Platforms that Support FCoE

```
[edit vlans vlan-name forwarding-options fip-security]
```

**NOTE:** The `beacon-period` configuration statement is in a different hierarchy on the original CLI than on the Enhanced Layer 2 Software (ELS) CLI.

**Release Information**

Statement introduced in Junos OS Release 12.2 for the QFX Series.
Statement introduced for the ELS CLI in Junos OS Release 13.2 for the QFX Series.

**Description**

Set the interval between periodic beacons. Beacons perform virtual link maintenance for VN_Ports in a way that is similar to FIP keepalive advertisements.

The ENode sends periodic beacons every 90 seconds on behalf of the VN_Port. Each received beacon resets the session timer for the virtual link connection to the other VN_Port. If the FCF does not receive a beacon before the beacon timer expires, the VN_Port is considered as "down" and the virtual link is terminated. The beacon timer expires in 2.5 times the configured beacon timer value.

**Options**

`milliseconds`—Time in milliseconds between beacons.

**Range:** 250 through 90000 milliseconds

**Default:** 8000 milliseconds

**Required Privilege Level**

storage—To view this statement in the configuration.

storage-control—To add this statement to the configuration.

**Related Documentation**

- Example: Configuring VN2VN_Port FIP Snooping (FCoE Hosts Directly Connected to the Same FCoE Transit Switch)
- Example: Configuring VN2VN_Port FIP Snooping (FCoE Hosts Directly Connected to Different FCoE Transit Switches)
- Example: Configuring VN2VN_Port FIP Snooping (FCoE Hosts Indirectly Connected Through an Aggregation Layer FCoE Transit Switch)
- Example: Configuring VN2VN_Port FIP Snooping (FCoE Hosts Directly Connected to the Same FCoE Transit Switch) on page 147
- Example: Configuring VN2VN_Port FIP Snooping (FCoE Hosts Directly Connected to Different FCoE Transit Switches) on page 152
### code-points (Application Maps)

**Syntax**

```
code-points [ aliases ] [ bit-patterns ];
```

**Hierarchy Level**

```
[edit policy-options application-maps application-map-name application application-name]
```

**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**

- `aliases`—Name of the alias or aliases.
- `bit-patterns`—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**

- Configuring an Application Map for DCBX Application Protocol TLV Exchange on page 191
- Example: Configuring DCBX Application Protocol TLV Exchange on page 105
- Example: Configuring DCBX to Support an iSCSI Application
- Understanding DCBX Application Protocol TLV Exchange on page 97
- Understanding DCBX Application Protocol TLV Exchange on EX Series Switches
**dcbx**

Syntax

```
dcbx {
  disable;
  interface (interface-name | all) {
    disable;
    application-map application-map-name;
    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;
  }
}
```

Hierarchy Level  [edit protocols]

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.
**mode** and **recommendation-tlv** statements introduced in Junos OS Release 12.2 for the QFX Series.

Description  Configure DCBX properties.

Options  The 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

- show dcbx neighbors on page 249
- Understanding DCB Features and Requirements on page 16
- Configuring DCBX Autonegotiation on page 186
- Understanding DCB Features and Requirements on EX Series Switches
- Disabling DCBX to Disable PFC Autonegotiation on EX Series Switches (CLI Procedure)
dcbx-version

Syntax  dcbx-version (auto-negotiate | ieee-dcbx | dcbx-version-1.01);

Hierarchy Level  [edit protocols dcbx interface (all | interface-name)]

Release Information  Statement introduced in Junos OS Release 12.2 for the QFX Series.

Description  Set the DCBX version for the specified interface or interfaces.

QFX3500 switches come up in IEEE DCBX mode and then autonegotiate with the connected peer to set the DCBX version.

QFabric system Node devices come up using DCBX version 1.01, and then autonegotiate with the connected peer to set the DCBX mode.

Default  The default DCBX mode is autonegotiation.

Options  auto-negotiate—Automatically negotiate the DCBX version with the connected peer.

ieee-dcbx—Force the interface to use IEEE DCBX mode, regardless of the peer configuration.

dcbx-version-1.01—Force the interface to use version 1.01 DCBX mode, regardless of the peer configuration.

Required Privilege  routing—to view this statement in the configuration.

Level  routing-control—to add this statement to the configuration.

Related Documentation  • show dcbx neighbors on page 249

• Configuring DCBX Autonegotiation on page 186

• Understanding DCBX on page 87
destination-port (Applications)

**Syntax**

destination-port \textit{port-value};

**Hierarchy Level**

[edit applications \texttt{application} application-name]

**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 \texttt{protocol} to identify an application type. The Internet Assigned Numbers Authority (IANA) assigns port numbers. See the IANA Service Name and Transport Protocol Port Number Registry at http://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xml for a list of assigned port numbers.

\begin{quote}
\textbf{NOTE: To create an application for iSCSI, use the protocol tcp with the destination port number 3260.}
\end{quote}

**Options**

\textit{port-value}—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**

- Defining an Application for DCBX Application Protocol TLV Exchange on page 189
- Example: Configuring DCBX Application Protocol TLV Exchange on page 105
- Example: Configuring DCBX to Support an iSCSI Application
- Understanding DCBX Application Protocol TLV Exchange on page 97
- Understanding DCBX Application Protocol TLV Exchange on EX Series Switches
disable (DCBX)

Syntax disable

Hierarchy Level [edit protocols dcbx]

[edit protocols dcbx interface interface-name]

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
• Configuring DCBX Autonegotiation on page 186
• Disabling DCBX to Disable PFC Autonegotiation on EX Series Switches (CLI Procedure)
• Understanding DCB Features and Requirements on page 16
• Understanding DCB Features and Requirements on EX Series Switches
**enhanced-transmission-selection**

**Syntax**
```
enhanced-transmission-selection {
    no-auto-negotiation;
    no-recommendation-tlv;
    recommendation-tlv {
        no-auto-negotiation;
    }
}
```

**Hierarchy Level**
```
[edit protocols dcbx interface interface-name]
```

**Release Information**
Statement introduced in Junos OS Release 11.1 for the QFX Series.

**Description**
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.

Disabling ETS autonegotiation stops the QFX Series from advertising the ETS Configuration TLV and the ETS Recommendation TLV.

Disabling the ETS recommendation TLV stops the QFX Series from advertising the ETS Recommendation TLV, but the ETS Configuration TLV is still advertised.

**Options**
- `no-auto-negotiation`—Disable automatic negotiation of ETS (Configuration TLV and Recommendation TLV)
- `no-recommendation-tlv`—Disable automatic negotiation of the ETS Recommendation TLV
- `recommendation-tlv`—Enable automatic negotiation of 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**
- [show dcbx neighbors on page 249](#)
- [Configuring DCBX Autonegotiation on page 186](#)
- [Example: Configuring CoS Hierarchical Port Scheduling (ETS)](#)
- [Understanding DCB Features and Requirements on page 16](#)
ether-type

**Syntax**
ether-type *ether-type*;

**Hierarchy Level**
[edit applications application application-name]

**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](http://standards.ieee.org/develop/regauth/ethertype/eth.txt) for a list of Institute of Electrical and Electronics Engineers (IEEE) EtherTypes.

---

**NOTE:** To create a FIP application, use the EtherType 0x8914.

---

**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**
- [Defining an Application for DCBX Application Protocol TLV Exchange on page 189](#)
- [Example: Configuring DCBX Application Protocol TLV Exchange on page 105](#)
- [Understanding DCBX Application Protocol TLV Exchange on page 97](#)
examine-vn2vn

Syntax

```
examine-vn2vn { 
  beacon-period milliseconds; 
}
```

Hierarchy Level

Original CLI

```
[edit ethernet-switching options secure-access-port vlan (all | vlan-name) examine-fip]
ELS CLI for Platforms that Support FCoE
[edit vlans vlan-name forwarding-options fip-security]
```

NOTE: The `examine-vn2vn` configuration statement is in a different hierarchy on the original CLI than on the Enhanced Layer 2 Software (ELS) CLI.

Release Information

Statement introduced in Junos OS Release 12.2 for the QFX Series.
Statement introduced for the ELS CLI in Junos OS Release 13.2 for the QFX Series.

Description

Enable VN_Port to VN_Port (VN2VN) FIP snooping on a specified VLAN. The VLAN must be a dedicated FCoE VLAN that transports only FCoE traffic. A VLAN cannot support VN2VN FIP snooping and VN_Port to VF_Port FIP snooping (VN2VF) simultaneously. Configure separate VLANs for VN2VN FIP snooping and VN2VF FIP snooping.

When you enable VN2VN FIP snooping on a VLAN, the VN2VF session filters are removed and the all existing VN2VF sessions are terminated.

The remaining statement is explained separately.

Required Privilege

- **routing**—To view this statement in the configuration.
- **routing-control**—To add this statement to the configuration.

Related Documentation

- Example: Configuring VN2VN_Port FIP Snooping (FCoE Hosts Directly Connected to the Same FCoE Transit Switch)
- Example: Configuring VN2VN_Port FIP Snooping (FCoE Hosts Directly Connected to Different FCoE Transit Switches)
- Example: Configuring VN2VN_Port FIP Snooping (FCoE Hosts Indirectly Connected Through an Aggregation Layer FCoE Transit Switch)
- Example: Configuring VN2VN_Port FIP Snooping (FCoE Hosts Directly Connected to the Same FCoE Transit Switch) on page 147
- Example: Configuring VN2VN_Port FIP Snooping (FCoE Hosts Directly Connected to Different FCoE Transit Switches) on page 152
- Example: Configuring VN2VN_Port FIP Snooping (FCoE Hosts Indirectly Connected Through an Aggregation Layer FCoE Transit Switch) on page 159
### fc-map

**Syntax**

`fc-map fc-map-value;`

**Hierarchy Level**

Original CLI

```
[edit ethernet-switching options secure-access-port vlan (all | vlan-name) examine-fip]
ELS CLI for Platforms that Support FCoE
```

```
[edit vlans vlan-name forwarding-options fip-security]
```

**NOTE:** The `fc-map` configuration statement is in a different hierarchy on the original CLI than on the Enhanced Layer 2 Software (ELS) CLI.

**QFX Series that Support FCoE-FC Gateway Configuration**

```
[edit fc-fabrics fc-fabric-name protocols fip]
```

**Release Information**

Statement introduced in Junos OS Release 10.4 for EX Series switches.

Statement introduced in Junos OS Release 11.1 for the QFX Series.

Statement introduced for the ELS CLI in Junos OS Release 13.2 for the QFX Series.

**Description**

Set the FCoE mapped address prefix (FC-MAP) value for the FCoE VLAN to match the FC switch (or FCoE forwarder) FC-MAP value for the FC fabric. The FC-MAP value is a unique MAC address prefix an FC switch uses to identify FCoE traffic for a given FC fabric (traffic on a particular FCoE VLAN).

You can configure the FC-MAP value or use the default value. The default FC-MAP value is different for VN_Port to VF_Port (VN2VF_Port) FIP snooping (0x0EFC00) than for VN_Port to VN_Port (VN2VN_Port) FIP snooping.

The FC switch provides the FC-MAP value to FCoE nodes (ENodes) in the FIP discovery advertisement message. If the EX Series switch or the QFX Series FCoE VLAN FC-MAP value does not match the FC switch FC-MAP value, neither device discovers the FC switch on that VLAN, and the ENodes on that VLAN cannot access the FC switch. The FC switch accepts only FCoE traffic that uses the correct FC-MAP value as part of the VN_Port MAC address.

When the QFX Series acts as an FCoE-FC gateway, the FC-MAP value for the gateway and the FCoE devices must match the FC switch FC-MAP value in order to communicate with the FC switch.

**NOTE:** Changing the FC-MAP value causes all logins to drop and forces the ENodes to log in again.

**Options**

`fc-map-value`—FC-MAP value, hexadecimal value preceded by “0x”.

---

Copyright © 2014, Juniper Networks, Inc.
Range: 0x0EFC00 through 0x0EFCFF
Default: 0x0EFC00 for VN2VF_Port FIP snooping 0x0EFD00 for VN2VN_Port FIP snooping

Required Privilege Level
routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Documentation
• examine-fip
• show fip snooping on page 326
• Example: Configuring an FCoE Transit Switch
• Configuring VN2VF_Port FIP Snooping and FCoE Trusted Interfaces on an FCoE Transit Switch on page 179
**fcoe-trusted**

**Syntax**

```
fcoe-trusted;
```

**Hierarchy Level**

Original CLI

```
[edit ethernet-switching-options secure-access-port interface interface-name]
ELS CLI for Platforms that Support FCoE

[edit vlans vlan-name forwarding-options fip-security interface interface-name]
```

**NOTE:** The `fcoe-trusted` configuration statement is in a different hierarchy on the original CLI than on the Enhanced Layer 2 Software (ELS) CLI.

**QFX Series that Support FCoE-FC Gateway Configuration**

```
[edit fc-fabrics fc-fabric-name protocols fip]
```

**Release Information**

Statement introduced in Junos OS Release 10.4 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced for the FC fabric in Junos OS Release 11.3 for the QFX Series.
Statement introduced for the ELS CLI in Junos OS Release 13.2 for the QFX Series.

**Description**

Configure the specified 10-Gigabit Ethernet interface to trust Fibre Channel over Ethernet (FCoE) traffic. If an interface is connected to another switch such as an FCoE forwarder (FCF) or a transit switch, you can configure the interface as trusted so that the interface forwards FCoE traffic from the switch to the FCoE devices without installing FIP snooping filters.

(QFX Series FCoE-FC gateway) Configure the specified local Fibre Channel fabric to trust FCoE traffic on all ports in the fabric. Changing the fabric ports from untrusted to trusted removes any existing FIP snooping filters from the ports. Changing the fabric ports from trusted to untrusted by removing the `fcoe-trusted` configuration from the fabric forces all of the FCoE sessions on those ports to log out so that when the ENodes and VN_Ports log in again, the switch can build the appropriate FIP snooping filters.

**Required Privilege Level**

- routing—To view this statement in the configuration.
- routing-control—To add this statement to the configuration.

**Related Documentation**

- [show fip snooping on page 326](#)
- [Example: Configuring an FCoE Transit Switch](#)
  - Configuring VN2VF_Port FIP Snooping and FCoE Trusted Interfaces on an FCoE Transit Switch on page 179
  - Configuring VN2VF_Port FIP Snooping and FCoE Trusted Interfaces on an FCoE Transit Switch on page 179
interface (DCBX)

Syntax

```plaintext
interface (interface-name | all) {
  disable;
  application-map application-map-name:
  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;
  }
}
```

Hierarchy Level

[edit protocols dcbx]

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 11.3 for the EX Series switches.
Mode and recommendation-tlv statements introduced in Junos OS Release 12.2 for the QFX Series.

Description

Configure DCBX properties on an interface.

Options

`interface-name`—Name of the interface.

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

- show dcbx neighbors on page 249
- Configuring DCBX Autonegotiation on page 186
- Example: Configuring DCBX to Support an iSCSI Application
- Understanding DCB Features and Requirements on page 16
- Understanding DCB Features and Requirements on EX Series Switches
- Understanding DCBX Application Protocol TLV Exchange on EX Series Switches
### no-recommendation-tlv

<table>
<thead>
<tr>
<th><strong>Syntax</strong></th>
<th>no-recommendation-tlv;</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Hierarchy Level</strong></td>
<td>[edit protocols dcbx interface interface-name enhanced-transmission-selection]</td>
</tr>
<tr>
<td><strong>Release Information</strong></td>
<td>Statement introduced in Junos OS Release 12.2 for the QFX Series.</td>
</tr>
<tr>
<td><strong>Description</strong></td>
<td>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.)</td>
</tr>
<tr>
<td><strong>Default</strong></td>
<td>DCBX-enabled interfaces send the ETS recommendation TLV unless it is disabled.</td>
</tr>
<tr>
<td><strong>Required Privilege Level</strong></td>
<td>routing—to view this statement in the configuration. routing-control—to add this statement to the configuration.</td>
</tr>
<tr>
<td><strong>Related Documentation</strong></td>
<td>• show dcbx neighbors on page 249 • Configuring DCBX Autonegotiation on page 186</td>
</tr>
</tbody>
</table>
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.

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 189
- Example: Configuring DCBX Application Protocol TLV Exchange on page 105
- Example: Configuring DCBX to Support an iSCSI Application
- Understanding DCBX Application Protocol TLV Exchange on page 97
- Understanding DCBX Application Protocol TLV Exchange on EX Series Switches
priority-flow-control

Syntax

```plaintext
priority-flow-control {
    no-auto-negotiation;
}
```

Hierarchy Level

[edit protocols dcbx interface (all | interface-name)]

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

- show dcbx neighbors on page 249
- Configuring CoS PFC (Congestion Notification Profiles)
- Configuring Priority-Based Flow Control for an EX Series Switch (CLI Procedure)
- Configuring DCBX Autonegotiation on page 186
- Example: Configuring CoS PFC for FCoE Traffic on page 115
- Understanding Data Center Bridging Capability Exchange Protocol for EX Series Switches
- Understanding Priority-Based Flow Control
- Understanding DCB Features and Requirements on page 16
**protocol (Applications)**

**Syntax**  
protocol (tcp | udp);

**Hierarchy Level**  
[edit applications application application-name]

**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.

---

**NOTE:** To create an application for iSCSI, use the protocol tcp with the destination port number 3260.

---

**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**  
- Defining an Application for DCBX Application Protocol TLV Exchange on page 189  
- Example: Configuring DCBX Application Protocol TLV Exchange on page 105  
- *Example: Configuring DCBX to Support an iSCSI Application*  
- Understanding DCBX Application Protocol TLV Exchange on page 97  
- *Understanding DCBX Application Protocol TLV Exchange on EX Series Switches*
recommendation-tlv

Syntax

recommendation-tlv {
   no-auto-negotiation;
}

Hierarchy Level

[edit protocols dcbx interface interface-name 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

routing—To view this statement in the configuration.

Level

routing-control—To add this statement to the configuration.

Related Documentation

• show dcbx neighbors on page 249
• Configuring DCBX Autonegotiation on page 186
CHAPTER 9

Configuration Statements (ELS CLI for Platforms that Support FCoE Only)

- examine-vn2vf on page 220
- fip-security on page 221
- interface (FIP Snooping) on page 222
### examine-vn2vf

<table>
<thead>
<tr>
<th>Syntax</th>
<th>examine-vn2vf</th>
</tr>
</thead>
</table>

**Hierarchy Level**

```
[edit vlans vlan-name forwarding-options flip-security]
```

**Release Information**

Statement introduced for the ELS CLI in Junos OS Release 13.2 for the QFX Series.

**Description**

NOTE: This statement supports the Enhanced Layer 2 Software (ELS) CLI. If your switch runs the original (non-ELS) software, see `examine-fip`. For ELS details, see [Getting Started with Enhanced Layer 2 Software](#).

Enable VN_Port to VF_Port (VN2VF_Port) FIP snooping on the specified VLAN. Ensure that the VLAN is a dedicated FCoE VLAN that transports only FCoE traffic.

If the switch also performs VN_Port to VN_Port (VN2VN_Port) FIP snooping, ensure that the VN2VN_Port traffic is on a different VLAN than the VN2VF_Port traffic. You cannot mix VN2VF_Port and VN2VN.Port traffic in the same VLAN, so you must use separate VLANs for VN2VF_Port and VN2VN.Port traffic.

**Required Privilege Level**

- routing—To view this statement in the configuration.
- routing-control—To add this statement to the configuration.

**Related Documentation**

- [examine-vn2vn on page 208](#)
- [Understanding VN_Port to VF_Port FIP Snooping on an FCoE Transit Switch](#)
- [Understanding FCoE Transit Switch Functionality](#)
- [Configuring VN2VF.Port FIP Snooping and FCoE Trusted Interfaces on an FCoE Transit Switch](#)
**fip-security**

**Syntax**

```plaintext
fip-security {
  examine-vn2vf;
  examine-vn2vn {
    beacon-period milliseconds;
  }
  fc-map fc-map-value;
  interface interface-name {
    (fcoe-trusted | no-fcoe-trusted;)
  }
}
```

**Hierarchy Level**

[edit vlans vlan-name forwarding-options]

**Release Information**

Statement introduced for the ELS CLI in Junos OS Release 13.2 for the QFX Series.

**Description**

NOTE: This statement supports the Enhanced Layer 2 Software (ELS) CLI. If your switch runs the original (non-ELS) software, see `examine-fip`. For ELS details, see *Getting Started with Enhanced Layer 2 Software*.

Configure FIP snooping and FCoE interface properties.

**Required Privilege**

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

**Related Documentation**

- Understanding VN_Port to VF_Port FIP Snooping on an FCoE Transit Switch on page 38
- Understanding VN_Port to VN_Port FIP Snooping on an FCoE Transit Switch on page 45
- Understanding FCoE Transit Switch Functionality on page 26
- Configuring VN2VF_Port FIP Snooping and FCoE Trusted Interfaces on an FCoE Transit Switch on page 179
- Enabling VN2VN_Port FIP Snooping and Configuring the Beacon Period on an FCoE Transit Switch on page 182
interface (FIP Snooping)

Syntax

```
interface interface-name {
    (fcoe-trusted| no-fcoe-trusted);
}
```

Hierarchy Level

```
[edit vlans vlan-name forwarding-options fip-security]
```

Release Information

Statement introduced for the ELS CLI in Junos OS Release 13.2 for the QFX Series.

Description

NOTE: This statement supports the Enhanced Layer 2 Software (ELS) CLI. If your switch runs the original (non-ELS) software, see interface (Secure Access Port) for how to specify an interface to configure as FCoE trusted or FCoE untrusted. For ELS details, see Getting Started with Enhanced Layer 2 Software.

Specify an interface to set as FCoE trusted or as FCoE untrusted. Configure interfaces that connect to other switches as trusted interfaces. Configure interfaces that connect directly to FCoE devices as untrusted interfaces and enabled FIP snooping on the untrusted interfaces to prevent unauthorized access to the storage network.

Required Privilege Level

- routing—To view this statement in the configuration.
- routing-control—To add this statement to the configuration.

Related Documentation

- Configuring VN2VF_Port FIP Snooping and FCoE Trusted Interfaces on an FCoE Transit Switch on page 179
- Understanding FCoE Transit Switch Functionality on page 26
PART 3

Administration

- Operational Commands on page 225
CHAPTER 10

Operational Commands

- clear fibre-channel fc2 statistics
- clear fibre-channel fip enode
- clear fibre-channel fip statistics
- clear fibre-channel fip vn-port
- clear fibre-channel flogi statistics
- clear fibre-channel proxy statistics
- clear fip snooping enode
- clear fip snooping statistics
- clear fip snooping vlan
- clear fip vlan-discovery statistics
- restart
- show dcbx
- show dcbx neighbors
- show fibre-channel fabric
- show fibre-channel fc2 sessions
- show fibre-channel fc2 statistics
- show fibre-channel fip
- show fibre-channel fip enode
- show fibre-channel fip fabric
- show fibre-channel fip fcf
- show fibre-channel fip interface
- show fibre-channel fip statistics
- show fibre-channel flogi fport
- show fibre-channel flogi nport
- show fibre-channel flogi statistics
- show fibre-channel interfaces
- show fibre-channel next-hops
- show fibre-channel routes
• show fibre-channel proxy fabric-state
• show fibre-channel proxy login-table
• show fibre-channel proxy np-port
• show fibre-channel proxy statistics
• show fip snooping
• show fip snooping enode
• show fip snooping fcf
• show fip snooping interface
• show fip snooping statistics
• show fip snooping vlan
• show fip vlan-discovery
• show route forwarding-table family fibre-channel
clear fibre-channel fc2 statistics

Syntax

clear fibre-channel fc2 statistics
<fabric fabric-name>

Release Information
Command introduced in Junos OS Release 11.1 for the QFX Series.

Description
Clear FC-2 (network layer) Fibre Channel statistics globally or on a specified Fibre Channel fabric.

Options

Required Privilege
view

Related Documentation
• show fibre-channel fc2 statistics on page 275
• show fibre-channel fc2 sessions on page 273

List of Sample Output
clear fibre-channel fc2 statistics on page 227

Sample Output
clear fibre-channel fc2 statistics

user@switch> clear fibre-channel fc2 statistics
clear fibre-channel fip enode

Syntax  
clear fibre-channel fip enode <enode-mac>

Release Information  
Command introduced in Junos OS Release 11.1 for the QFX Series.

Description  
Clear Fibre Channel over Ethernet (FCoE) node (ENode) information for a specified ENode. This operation deletes the ENode state from the switch database and from the FIP snooping firewall filters, which causes the ENode to lose the connection to the Fibre Channel (FC) fabric and to log in to the fabric again. If you clear an ENode, all VN_Ports associated with that ENode are also cleared and lose their connection to the FC fabric and must log in to the fabric again.

Options  

<enode-mac>—MAC address of the ENode.

Required Privilege  
view

Related Documentation  
• show fibre-channel fip enode on page 282
• clear fibre-channel fip statistics on page 229
• clear fibre-channel fip vn-port on page 230

List of Sample Output  
clear fibre-channel fip enode on page 228

Sample Output  
clear fibre-channel fip enode

user@switch> clear fibre-channel fip enode 00:10:94:00:00:02
clear fibre-channel fip statistics

Syntax

```
clear fibre-channel fip statistics
  <fabric fabric-name>
```

Release Information

Command introduced in Junos OS Release 11.1 for the QFX Series.

Description

Clear Fibre Channel over Ethernet (FCoE) initialization protocol (FIP) statistics.

Options

```
```

Required Privilege

view

Related Documentation

- show fibre-channel fip statistics on page 295
- show fibre-channel fip on page 277

List of Sample Output

clear fibre-channel fip statistics on page 229

Sample Output

clear fibre-channel fip statistics

```
user@switch> clear fibre-channel fip statistics
```
**clear fibre-channel fip vn-port**

**Syntax**
clear fibre-channel fip vn-port vn-port--mac

**Release Information**
Command introduced in Junos OS Release 11.1 for the QFX Series.

**Description**
Clear virtual N_Port (VN_Port) information for a specified VN_Port. This operation deletes the VN_Port state from the switch database and from the FIP snooping firewall filters, which causes the VN_Port to lose its connection to the Fibre Channel fabric and to log in to the fabric again. When you clear a VN_Port, other VN_Ports associated with the same Fibre Channel over Ethernet (FCoE) Node (ENode) are not affected and are not cleared.

**Options**
- **vn-port-mac**—MAC address of the VN_Port.

**Required Privilege Level**
view

**Related Documentation**
- show fibre-channel fip enode on page 282
- clear fibre-channel fip enode on page 228
- clear fibre-channel fip statistics on page 229

**List of Sample Output**
clear fibre-channel fip vn-port on page 230

**Sample Output**
clear fibre-channel fip vn-port

user@switch> clear fibre-channel fip vn-port 00:10:94:00:00:08
clear fibre-channel flogi statistics

Syntax

clear fibre-channel flogi statistics
    <fabric fabric-name>

Release Information

Command introduced in Junos OS Release 11.1 for the QFX Series.

Description

Clear fabric login (FLOGI) statistics globally or on a specified Fibre Channel fabric.

Options


Required Privilege

Level view

Related Documentation

- show fibre-channel flogi statistics on page 303
- show fibre-channel flogi fport on page 299
- show fibre-channel flogi nport on page 301

List of Sample Output

clear fibre-channel flogi statistics on page 231

Sample Output

clear fibre-channel flogi statistics

user@switch> clear fibre-channel flogi statistics
clear fibre-channel proxy statistics

Syntax

```
clear fibre-channel proxy statistics <fabric fabric-name>
```

Release Information
Command introduced in Junos OS Release 11.1 for the QFX Series.

Description
Clear Fibre Channel gateway statistics globally or on a specified Fibre Channel fabric.

Options


Required Privilege

- `view`

Related Documentation
- `show fibre-channel proxy statistics` on page 323
- `show fibre-channel proxy login-table` on page 317
- `show fibre-channel proxy np-port` on page 320

List of Sample Output
clear fibre-channel proxy statistics on page 232

Sample Output

clear fibre-channel proxy statistics

```
user@switch> clear fibre-channel proxy statistics
```
clear fip snooping enode

Syntax

```
clear fip snooping enode enode-mac
<vlan vlan-name>
```

Release Information

Command introduced in Junos OS Release 10.4 for EX Series switches.
Command introduced in Junos OS Release 11.1 for the QFX Series.

Description

Clear FIP snooping information for the specified FCoE Node (ENode) or (optionally) only on a specified VLAN. This operation deletes the ENode state from the switch database and from the FIP snooping firewall filters, which causes the ENode to lose its connection to the FCoE forwarder (FCF) and to log in to the FCF again.

Options

- **enode-mac**—MAC address of the ENode.
- **vlan vlan-name**—(Optional) Name of the VLAN.

Required Privilege Level

view

Related Documentation

- [show fip snooping enode](#) on page 331

List of Sample Output

```
clear fip snooping enode enode-mac on page 233
```

Sample Output

```
clear fip snooping enode enode-mac

user@switch> clear fip snooping enode 00:10:94:00:00:02
```
clear fip snooping statistics

Syntax

```
clear fip snooping statistics
  <vlan vlan-name>
```

Release Information

Command introduced in Junos OS Release 10.4 for EX Series switches.
Command introduced in Junos OS Release 11.1 for the QFX Series.

Description

Clear FIP snooping statistics globally or on a specified VLAN.

Required Privilege

`view`

Related Documentation

- `show fip snooping statistics` on page 341

List of Sample Output

`clear fip snooping statistics` on page 234

Sample Output

```
clear fip snooping statistics

user@switch> clear fip snooping statistics
```
clear fip snooping vlan

Syntax

clear fip snooping vlan vlan-name

Release Information

Command introduced in Junos OS Release 10.4 for EX Series switches. Command introduced in Junos OS Release 11.1 for the QFX Series.

Description

Clear FIP snooping information for the specified VLAN. This operation deletes all ENode and FCF information for the VLAN from the switch database and causes the ENodes to lose their connections to the FCFs. After clearing a VLAN, the switch relearns all of the FCFs and ENodes on the VLAN, and the ENodes must log in to the FCF again.

Options

vlan-name—Name of the VLAN.

Required Privilege

view

Related Documentation

• show fip snooping vlan on page 344

List of Sample Output

clear fip snooping vlan vlan-name on page 235

Sample Output

clear fip snooping vlan vlan-name

user@switch> clear fip snooping vlan fcoevlan1
clear fip vlan-discovery statistics

Syntax  
clear fip vlan-discovery statistics

Release Information  
Command introduced in Junos OS Release 12.1 for the QFX Series.

Description  
Clear FIP VLAN discovery statistics.

Required Privilege  
view

Level

Related Documentation  
•  show fip vlan-discovery on page 348

List of Sample Output  
clear fip vlan-discovery statistics on page 236

Sample Output

clear fip vlan-discovery statistics

user@switch>  clear fip vlan-discovery statistics
restart

List of Syntax

Syntax (ACX Series Routers) on page 237
Syntax (EX Series Switches) on page 237
Syntax (Routing Matrix) on page 238
Syntax (J Series Routing Platform) on page 238
Syntax (TX Matrix Routers) on page 238
Syntax (TX Matrix Plus Routers) on page 238
Syntax (MX Series Routers) on page 238
Syntax (J Series Routers) on page 239
Syntax (QFX Series) on page 239

Syntax

restart


<gracefully | immediately | soft>

Syntax (ACX Series Routers)

restart


Syntax (EX Series Switches)

restart

<autoinstallation | chassis-control | class-of-service | database-replication | dhcp | dhcp-service | diameter-service | dot1x-protocol | ethernet-link-fault-management | ethernet-switching | event-processing | firewall | general-authentication-service | interface-control | kernel-replication | l2-learning | lacp | license-service | link-management |
Syntax (Routing Matrix)

```
restart
<adaptive-services | audit-process | chassis-control | class-of-service | dhcp-service |
diameter-service | disk-monitoring | dynamic-flow-capture | ecc-error-logging |
event-processing | firewall | interface-control | ipsec-key-management |
| l2-learning | l2tp-service | lACP | link-management | mic-process | pgm | pic-services-logging | ppp | pppoe |
redundancy-interface-process | remote-operations | routing <logical-system logical-system-name> | sampling | service-deployment | snmp>
<all | all-lcc | lcc number>
<gracefully | immediately | soft>
```

Syntax (J Series Routing Platform)

```
restart
<adaptive-services | audit-process | chassis-control | class-of-service | dhcp | dialer-services |
dlsw | event-processing | firewall | interface-control | ipsec-key-management |
| isdn-signaling | l2-learning | l2tp-service | mic-process | network-access-service | pgm | ppp | pppoe | remote-operations | routing <logical-system logical-system-name> | sampling |
<service-deployment | snmp | usb-control | web-management>
<gracefully | immediately | soft>
```

Syntax (TX Matrix Routers)

```
restart
<adaptive-services | audit-process | chassis-control | class-of-service | dhcp-service |
diameter-service | disk-monitoring | dynamic-flow-capture | ecc-error-logging |
event-processing | firewall | interface-control | ipsec-key-management |
| l2-learning | l2tp-service | lACP | link-management | mic-process | pgm | pic-services-logging | ppp | pppoe |
redundancy-interface-process | remote-operations | routing <logical-system logical-system-name> | sampling | service-deployment | snmp | statistics-service>
<all-chassis | all-lcc | lcc number | scc>
<gracefully | immediately | soft>
```

Syntax (TX Matrix Plus Routers)

```
restart
<adaptive-services | ancpd-service | application-identification|audit-process |
auto-configuration | captive-portal-content-delivery | ce-l2tp-service | chassis-control |
class-of-service | clksyncd-service | database-replication | datapath-trace-service |
dhcp-service | diameter-service | disk-monitoring | dynamic-flow-capture |
ecc-error-logging | ethernet-connectivity-fault-management |
eternet-link-fault-management | event-processing | firewall |
general-authentication-service | gracefully | iccp-service | idp-policy | immediately |
| interface-control | ipsec-key-management | kernel-replication | l2-learning | l2cpd-service |
l2tp-service | l2tp-universal-edge | lACP | license-service | link-management |
| local-policy-decision-function | mac-validation | mic-process | mobile-ip | mountd-service |
mpls-traceroute | mspd | multicast-snooping | named-service | nfsd-service |
```
packet-triggered-subscribers | peer-selection-service | pgcp-service | pgm | pic-services-logging | pki-service | ppp | ppp-service | pppoe | protected-system-domain-service | redundancy-interface-process | remote-operations | root-system-domain-service | routing | routing <logical-system logical-system-name> | sampling | sbc-configuration-process | sdk-service | service-deployment | services | services pgcp gateway gateway-name | smtp | soft | static-subscribers | statistics-service | subscriber-management | subscriber-management-helper | tunnel-oam | usb-control | vrp | web-management>

<all-members>
<gracefully | immediately | soft>
<local>
<member member-id>

Syntax (J Series Routers)

restart
<adaptive-services | audit-process | chassis-control | class-of-service | dhcp | dhcp-service | dialer-services | diameter-service | dlsw | event-processing | firewall | interface-control | ipsec-key-management | isdn-signaling | l2ald | l2-learning | l2tp-service | mib-process | network-access-service | pgm | ppp | pppoe | remote-operations | routing <logical-system logical-system-name> | sampling | service-deployment | snmp | usb-control | web-management>
<gracefully | immediately | soft>

Syntax (QFX Series)

restart
<adaptive-services | audit-process | chassis-control | class-of-service | dialer-services | diameter-service | dlsw | ethernet-connectivity | event-processing | fibre-channel | firewall | general-authentication-service | igmp-host-services | interface-control | ipsec-key-management | isdn-signaling | l2ald | l2-learning | l2tp-service | mib-process | named-service | network-access-service | ntrace-process | pgm | ppp | pppoe | redundancy-interface-process | remote-operations | routing <logical-system logical-system-name> | sampling | secure-neighbor-discovery | service-deployment | snmp | usb-control | web-management>
<gracefully | immediately | soft>

Release Information

Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 9.0 for EX Series switches.
Command introduced in Junos OS Release 11.1 for the QFX Series.
Command introduced in Junos OS Release 12.2 for ACX Series routers.
Options added:

- **dynamic-flow-capture** in Junos OS Release 7.4.
- **dlsw** in Junos OS Release 7.5.
- **event-processing** in Junos OS Release 7.5.
- **ppp** in Junos OS Release 7.5.
- **l2ald** in Junos OS Release 8.0.
- **link-management** in Release 8.0.
- **pgcp-service** in Junos OS Release 8.4.
- **sbc-configuration-process** in Junos OS Release 9.5.
- **services pgcp gateway** in Junos OS Release 9.6.
- **sfc** and **all-sfc** for the TX Matrix Router in Junos OS Release 9.6.
Description  Restart a Junos OS process.

CAUTION: Never restart a software process unless instructed to do so by a customer support engineer. A restart might cause the router or switch to drop calls and interrupt transmission, resulting in possible loss of data.

Options

- none—Same as gracefully.
- adaptive-services—(Optional) Restart the configuration management process that manages the configuration for stateful firewall, Network Address Translation (NAT), intrusion detection services (IDS), and IP Security (IPsec) services on the Adaptive Services PIC.
- all-chassis—(TX Matrix and TX Matrix Plus routers only) (Optional) Restart the software process on all chassis.
- all-lcc—(TX Matrix and TX Matrix Plus routers only) (Optional) For a TX Matrix router, restart the software process on all T640 routers connected to the TX Matrix router. For a TX Matrix Plus router, restart the software process on all T1600 routers connected to the TX Matrix Plus router.
- all-members—(MX Series routers only) (Optional) Restart the software process for all members of the Virtual Chassis configuration.
- all-sfc—(TX Matrix Plus routers only) (Optional) For a TX Matrix Plus router, restart the software processes for the TX Matrix Plus router (or switch-fabric chassis).
- ancpd-service—(Optional) Restart the Access Node Control Protocol (ANCP) process, which works with a special Internet Group Management Protocol (IGMP) session to collect outgoing interface mapping events in a scalable manner.
- application-identification—(Optional) Restart the process that identifies an application using intrusion detection and prevention (IDP) to allow or deny traffic based on applications running on standard or nonstandard ports.
- audit-process—(Optional) Restart the RADIUS accounting process that gathers statistical data that can be used for general network monitoring, analyzing, and tracking usage patterns, for billing a user based on the amount of time or type of services accessed.
- auto-configuration—(Optional) Restart the Interface Auto-Configuration process.
- autoinstallation—(EX Series switches only) (Optional) Restart the autoinstallation process.
- captive-portal-content-delivery—(Optional) Restart the HTTP redirect service by specifying the location to which a subscriber’s initial Web browser session is redirected, enabling initial provisioning and service selection for the subscriber.
ce-l2tp-service—(M10, M10i, M7i, and MX Series routers only) (Optional) Restart the Universal Edge Layer 2 Tunneling Protocol (L2TP) process, which establishes L2TP tunnels and Point-to-Point Protocol (PPP) sessions through L2TP tunnels.

chassis-control—(Optional) Restart the chassis management process.

class-of-service—(Optional) Restart the class-of-service (CoS) process, which controls the router’s or switch’s CoS configuration.

clksyncd-service—(Optional) Restart the external clock synchronization process, which uses synchronous Ethernet (SyncE).

database-replication—(EX Series switches and MX Series routers only) (Optional) Restart the database replication process.

datapath-trace-service—(Optional) Restart the packet path tracing process.

dhcp—(J Series routers and EX Series switches only) (Optional) Restart the software process for a Dynamic Host Configuration Protocol (DHCP) server. A DHCP server allocates network IP addresses and delivers configuration settings to client hosts without user intervention.

dhcp-service—(Optional) Restart the Dynamic Host Configuration Protocol process.

dialer-services—(J Series routers and EX Series switches only) (Optional) Restart the ISDN dial-out process.

diameter-service—(Optional) Restart the diameter process.

disk-monitoring—(Optional) Restart disk monitoring, which checks the health of the hard disk drive on the Routing Engine.

dlsw—(J Series routers and QFX Series only) (Optional) Restart the data link switching (DLSw) service.

dot1x-protocol—(EX Series switches only) (Optional) Restart the port-based network access control process.

dynamic-flow-capture—(Optional) Restart the dynamic flow capture (DFC) process, which controls DFC configurations on Monitoring Services III PICs.

ecc-error-logging—(Optional) Restart the error checking and correction (ECC) process, which logs ECC parity errors in memory on the Routing Engine.

ethernet-connectivity-fault-management—(Optional) Restart the process that provides IEEE 802.1ag Operation, Administration, and Management (OAM) connectivity fault management (CFM) database information for CFM maintenance association end points (MEPs) in a CFM session.

ethernet-link-fault-management—(EX Series switches and MX Series routers only) (Optional) Restart the process that provides the OAM link fault management (LFM) information for Ethernet interfaces.
ethernet-switching—(EX Series switches only) (Optional) Restart the Ethernet switching process.

event-processing—(Optional) Restart the event process (eventd).

fibre-channel—(QFX Series only) (Optional) Restart the Fibre Channel process.

firewall—(Optional) Restart the firewall management process, which manages the firewall configuration and enables accepting or rejecting packets that are transiting an interface on a router or switch.

general-authentication-service—(EX Series switches and MX Series routers only) (Optional) Restart the general authentication process.

gracefully—(Optional) Restart the software process.

iccp-service—(Optional) Restart the Inter-Chassis Communication Protocol (ICCP) process.

idp-policy—(Optional) Restart the intrusion detection and prevention (IDP) protocol process.

immediately—(Optional) Immediately restart the software process.

interface-control—(Optional) Restart the interface process, which controls the router’s or switch’s physical interface devices and logical interfaces.

ipsec-key-management—(Optional) Restart the IPsec key management process.

isdn-signaling—(J Series routers and QFX Series only) (Optional) Restart the ISDN signaling process, which initiates ISDN connections.

kernel-replication—(Optional) Restart the kernel replication process, which replicates the state of the backup Routing Engine when graceful Routing Engine switchover (GRES) is configured.

l2-learning—(Optional) Restart the Layer 2 address flooding and learning process.

l2cpd-service—(Optional) Restart the Layer 2 Control Protocol process, which enables features such as Layer 2 protocol tunneling and nonstop bridging.

l2tp-service—(M10, M10i, M7i, and MX Series routers only) (Optional) Restart the Layer 2 Tunneling Protocol (L2TP) process, which sets up client services for establishing Point-to-Point Protocol (PPP) tunnels across a network and negotiating Multilink PPP if it is implemented.

l2tp-universal-edge—(MX Series routers only) (Optional) Restart the L2TP process, which establishes L2TP tunnels and PPP sessions through L2TP tunnels.

lacr—(Optional) Restart the Link Aggregation Control Protocol (LACP) process. LACP provides a standardized means for exchanging information between partner systems on a link to allow their link aggregation control instances to reach agreement on the identity of the LAG to which the link belongs, and then to move the link to that LAG,
and to enable the transmission and reception processes for the link to function in an orderly manner.

**lcc number**—(TX Matrix and TX Matrix Plus routers only) (Optional) For a TX Matrix router, restart the software process for a specific T640 router that is connected to the TX Matrix router. For a TX Matrix Plus router, restart the software process for a specific router that is connected to the TX Matrix Plus router. Replace *number* with the following values depending on the LCC configuration:

- 0 through 3, when T640 routers are connected to a TX Matrix router in a routing matrix.
- 0 through 3, when T1600 routers are connected to a TX Matrix Plus router in a routing matrix.
- 0 through 7, when T1600 routers are connected to a TX Matrix Plus router with 3D SIBs in a routing matrix.
- 0, 2, 4, or 6, when T4000 routers are connected to a TX Matrix Plus router with 3D SIBs in a routing matrix.

**license-service**—(EX Series switches only) (Optional) Restart the feature license management process.

**link-management**—(TX Matrix and TX Matrix Plus routers and EX Series switches only) (Optional) Restart the Link Management Protocol (LMP) process, which establishes and maintains LMP control channels.

**lldpd-service**—(EX Series switches only) (Optional) Restart the Link Layer Discovery Protocol (LLDP) process.

**local**—(MX Series routers only) (Optional) Restart the software process for the local Virtual Chassis member.

**local-policy-decision-function**—(Optional) Restart the process for the Local Policy Decision Function, which regulates collection of statistics related to applications and application groups and tracking of information about dynamic subscribers and static interfaces.

**mac-validation**—(Optional) Restart the Media Access Control (MAC) validation process, which configures MAC address validation for subscriber interfaces created on demux interfaces in dynamic profiles on MX Series routers.

**member member-id**—(MX Series routers only) (Optional) Restart the software process for a specific member of the Virtual Chassis configuration. Replace *member-id* with a value of 0 or 1.

**mib-process**—(Optional) Restart the Management Information Base (MIB) version II process, which provides the router's MIB II agent.

**mobile-ip**—(Optional) Restart the Mobile IP process, which configures Junos OS Mobile IP features.
mountd-service—(EX Series switches and MX Series routers only) (Optional) Restart the service for NFS mount requests.

mpls-traceroute—(Optional) Restart the MPLS Periodic Traceroute process.

mspd—(Optional) Restart the Multiservice process.

multicast-snooping—(EX Series switches and MX Series routers only) (Optional) Restart the multicast snooping process, which makes Layer 2 devices, such as VLAN switches, aware of Layer 3 information, such as the media access control (MAC) addresses of members of a multicast group.

named-service—(Optional) Restart the DNS Server process, which is used by a router or a switch to resolve hostnames into addresses.

network-access-service—(J Series routers and QFX Series only) (Optional) Restart the network access process, which provides the router’s Challenge Handshake Authentication Protocol (CHAP) authentication service.

nfsd-service—(Optional) Restart the Remote NFS Server process, which provides remote file access for applications that need NFS-based transport.

packet-triggered-subscribers—(Optional) Restart the packet-triggered subscribers and policy control (PTSP) process, which allows the application of policies to dynamic subscribers that are controlled by a subscriber termination device.

peer-selection-service—(Optional) Restart the Peer Selection Service process.

pgcp-service—(Optional) Restart the pgcpd service process running on the Routing Engine. This option does not restart pgcpd processes running on mobile station PICs. To restart pgcpd processes running on mobile station PICs, use the services pgcp gateway option.

pgm—(Optional) Restart the process that implements the Pragmatic General Multicast (PGM) protocol for assisting in the reliable delivery of multicast packets.

pic-services-logging—(Optional) Restart the logging process for some PICs. With this process, also known as fsad (the file system access daemon), PICs send special logging information to the Routing Engine for archiving on the hard disk.

pki-service—(Optional) Restart the PKI Service process.

ppp—(Optional) Restart the Point-to-Point Protocol (PPP) process, which is the encapsulation protocol process for transporting IP traffic across point-to-point links.

ppp-service—(Optional) Restart the Universal Edge PPP process, which is the encapsulation protocol process for transporting IP traffic across Universal Edge routers.

pppoe—(Optional) Restart the Point-to-Point Protocol over Ethernet (PPPoE) process, which combines PPP that typically runs over broadband connections with the Ethernet link-layer protocol that allows users to connect to a network of hosts over a bridge or access concentrator.
**protected-system-domain-service**—(Optional) Restart the Protected System Domain (PSD) process.

**redundancy-interface-process**—(Optional) Restart the ASP redundancy process.

**remote-operations**—(Optional) Restart the remote operations process, which provides the ping and traceroute MIBs.

**root-system-domain-service**—(Optional) Restart the Root System Domain (RSD) service.

**routing**—(ACX Series routers, QFX Series, EX Series switches, and MX Series routers only) (Optional) Restart the routing protocol process.

**routing logical-system logical-system-name**—(Optional) Restart the routing protocol process, which controls the routing protocols that run on the router or switch and maintains the routing tables. Optionally, restart the routing protocol process for the specified logical system only.

**sampling**—(Optional) Restart the sampling process, which performs packet sampling based on particular input interfaces and various fields in the packet header.

**sbc-configuration-process**—(Optional) Restart the session border controller (SBC) process of the border signaling gateway (BSG).

**scc**—(TX Matrix routers only) (Optional) Restart the software process on the TX Matrix router (or switch-card chassis).

**sdk-service**—(Optional) Restart the SDK Service process, which runs on the Routing Engine and is responsible for communications between the SDK application and Junos OS. Although the SDK Service process is present on the router, it is turned off by default.

**secure-neighbor-discovery**—(QFX Series, EX Series switches, and MX Series routers only) (Optional) Restart the secure Neighbor Discovery Protocol (NDP) process, which provides support for protecting NDP messages.

**sfc number**—(TX Matrix Plus routers only) (Optional) Restart the software process on the TX Matrix Plus router (or switch-fabric chassis). Replace *number* with 0.

**service-deployment**—(Optional) Restart the service deployment process, which enables Junos OS to work with the Session and Resource Control (SRC) software.

**services**—(Optional) Restart a service.

**services pgcp gateway gateway-name**—(Optional) Restart the pgcpd process for a specific border gateway function (BGF) running on an MS-PIC. This option does not restart the pgcpd process running on the Routing Engine. To restart the pgcpd process on the Routing Engine, use the pgcp-service option.

**sflow-service**—(EX Series switches only) (Optional) Restart the flow sampling (sFlow technology) process.
**snmp**—(Optional) Restart the SNMP process, which enables the monitoring of network devices from a central location and provides the router’s or switch’s SNMP master agent.

**soft**—(Optional) Reread and reactivate the configuration without completely restarting the software processes. For example, BGP peers stay up and the routing table stays constant. Omitting this option results in a graceful restart of the software process.

**static-subscribers**—(Optional) Restart the static subscribers process, which associates subscribers with statically configured interfaces and provides dynamic service activation and activation for these subscribers.

**statistics-service**—(Optional) Restart the process that manages the Packet Forwarding Engine statistics.

**subscriber-management**—(Optional) Restart the Subscriber Management process.

**subscriber-management-helper**—(Optional) Restart the Subscriber Management Helper process.

**tunnel-oamd**—(Optional) Restart the Tunnel OAM process, which enables the Operations, Administration, and Maintenance of Layer 2 tunneled networks. Layer 2 protocol tunneling (L2PT) allows service providers to send Layer 2 PDUs across the provider’s cloud and deliver them to Juniper Networks EX Series Ethernet Switches that are not part of the local broadcast domain.

**usb-control**—(J Series routers and MX Series routers only) (Optional) Restart the USB control process.

**vrrp**—(ACX Series routers, EX Series switches, and MX Series routers only) (Optional) Restart the Virtual Router Redundancy Protocol (VRRP) process, which enables hosts on a LAN to make use of redundant routing platforms on that LAN without requiring more than the static configuration of a single default route on the hosts.

**web-management**—(J Series routers, QFX Series, EX Series switches, and MX Series routers only) (Optional) Restart the Web management process.

**Required Privilege Level**

- reset

**Related Documentation**

- Overview of Junos OS CLI Operational Mode Commands

**List of Sample Output**

- restart interfaces on page 246

**Output Fields**

When you enter this command, you are provided feedback on the status of your request.

**Sample Output**

```
restart interfaces

user@host> restart interfaces
interfaces process terminated
interfaces process restarted
```
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

```
view
```

Related Documentation

- show dcbx neighbors on page 249
- Configuring DCBX Autonegotiation on page 186

Output Fields
Table 19 on page 248 lists the output fields for the show dcbx command. Output fields are listed in the approximate order in which they appear.

Table 19: show dcbx output fields

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DCBX</td>
<td>Status of DCBX on the switch or for the specified interface:</td>
</tr>
<tr>
<td></td>
<td>- Enabled—DCBX is enabled on the switch or on the specified interface</td>
</tr>
<tr>
<td></td>
<td>- Disabled—DCBX is disabled on the switch or on the specified interface</td>
</tr>
<tr>
<td>Interface</td>
<td>Name of the interface</td>
</tr>
</tbody>
</table>

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**

**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**
view

**Related Documentation**
- Configuring DCBX Autonegotiation on page 186
- Example: Configuring DCBX Application Protocol TLV Exchange on page 105
- Example: Configuring an FCoE Transit Switch
- Example: Configuring DCBX to Support an iSCSI Application
- Understanding DCB Features and Requirements on page 16
- Understanding Data Center Bridging Capability Exchange Protocol for EX Series Switches
dcbx on page 202

**List of Sample Output**
- show dcbx neighbors interface (QFX Series, DCBX Version 1.01 Mode) on page 262
- show dcbx neighbors interface (QFX Series, IEEE DCBX Mode) on page 264
- show dcbx neighbors terse (QFX Series) on page 266
- show dcbx neighbors (EX4500 Switch: FCoE Interfaces on Both Local and Peer with PFC Configured Compatibly) on page 266
- show dcbx neighbors (EX4500 Switch: DCBX Interfaces on Local and Peer Are Configured Compatibly with iSCSI Application) on page 267
- show dcbx neighbors (EX4500 Switch: Includes ETS) on page 268

**Output Fields**
Table 20 on page 249 lists the output fields for the `show dcbx neighbors` command. Output fields are listed in the approximate order in which they appear.

**Table 20: show dcbx neighbors Output Fields**

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interface</td>
<td>Name of the interface.</td>
</tr>
</tbody>
</table>
Table 20: show dcbx neighbors Output Fields (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parent Interface</td>
<td>Name of the link aggregation group (LAG) interface to which the DCBX interface belongs.</td>
</tr>
<tr>
<td>Active-application-map</td>
<td>Name of the application map applied to the interface.</td>
</tr>
<tr>
<td>Protocol-Mode (QFX Series)</td>
<td>DCBX protocol mode the interface uses:</td>
</tr>
<tr>
<td></td>
<td>• IEEE DCBX Version—The interface uses IEEE DCBX mode.</td>
</tr>
<tr>
<td></td>
<td>• DCBX Version 1.01—The interface uses DCBX version 1.01.</td>
</tr>
<tr>
<td></td>
<td>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.</td>
</tr>
<tr>
<td>Protocol-State (DCBX Version 1.01 only)</td>
<td>DCBX protocol state synchronization status:</td>
</tr>
<tr>
<td></td>
<td>• 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.</td>
</tr>
<tr>
<td></td>
<td>• 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.</td>
</tr>
<tr>
<td>Local-Advertisement (DCBX Version 1.01 only)</td>
<td>Status of advertisements that the local interface sends to the peer.</td>
</tr>
<tr>
<td>Operational version</td>
<td>Version of the DCBX standard used.</td>
</tr>
<tr>
<td>sequence-number</td>
<td>Number of state change messages sent to the peer.</td>
</tr>
<tr>
<td></td>
<td>If the interface Protocol-State value is in-sync, this number should match the acknowledge-id number in the Peer-Advertisement section.</td>
</tr>
<tr>
<td></td>
<td>If the interface Protocol-State value is ack-pending, this number does not match the acknowledge-id number in the Peer-Advertisement section.</td>
</tr>
<tr>
<td>acknowledge-id</td>
<td>Number of acknowledge messages received from the peer.</td>
</tr>
<tr>
<td></td>
<td>If the Protocol-State value is in-sync, this number should match the sequence-number value in the Peer-Advertisement section.</td>
</tr>
<tr>
<td></td>
<td>If the Protocol-State value is ack-pending, this number does not match the sequence-number value in the Peer-Advertisement section.</td>
</tr>
</tbody>
</table>
### Table 20: show dcbx neighbors Output Fields (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Peer-Advertisement</strong></td>
<td>(DCBX Version 1.01 only) Status of advertisements that the peer sends to the local interface.</td>
</tr>
<tr>
<td><strong>Operational version</strong></td>
<td>Version of the DCBX standard used.</td>
</tr>
<tr>
<td><strong>sequence-number</strong></td>
<td>Number of state change messages the peer sent to the local interface. If this number matches the <strong>acknowledge-id</strong> number in the <strong>Local-Advertisement</strong> field, this indicates that the local interface has acknowledged all of the peer's state change messages and is synchronized. If this number does not match the <strong>acknowledge-id</strong> number in the <strong>Local-Advertisement</strong> field, this indicates that the peer has not yet received an acknowledgment for a state change message from the local interface.</td>
</tr>
<tr>
<td><strong>acknowledge-id</strong></td>
<td>Number of acknowledge messages the peer has received from the local interface. If this number matches the <strong>sequence-number</strong> value in the <strong>Local-Advertisement</strong> field, this indicates that the peer has acknowledged all of the local interface's state change messages and is in synchronization. If this number does not match the <strong>sequence-number</strong> value in the <strong>Local-Advertisement</strong> field, this indicates that the peer has not yet sent an acknowledgment for a state change message from the local interface.</td>
</tr>
</tbody>
</table>
Table 20: show dcbx neighbors Output Fields (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feature: PFC</td>
<td>Priority-based flow control (PFC) feature DCBX state information.</td>
</tr>
<tr>
<td>Protocol-State</td>
<td>(DCBX Version 1.01 only)</td>
</tr>
<tr>
<td></td>
<td>DCBX protocol state synchronization status:</td>
</tr>
<tr>
<td></td>
<td>• <strong>ack-pending</strong>—The local interface has not yet received an acknowledge message</td>
</tr>
<tr>
<td></td>
<td>from the peer to indicate that the peer received a PFC state change message sent</td>
</tr>
<tr>
<td></td>
<td>by the local interface.</td>
</tr>
<tr>
<td></td>
<td>• <strong>in-sync</strong>—The local interface received an acknowledge message from the peer</td>
</tr>
<tr>
<td></td>
<td>to indicate that the peer received a PFC state change message sent by the local</td>
</tr>
<tr>
<td></td>
<td>interface.</td>
</tr>
<tr>
<td></td>
<td>• <strong>not-applicable</strong>—PFC autonegotiation is disabled.</td>
</tr>
<tr>
<td>Operational State</td>
<td>(DCBX Version 1.01 only)</td>
</tr>
<tr>
<td></td>
<td>Operational state of the feature: <strong>enabled</strong> or <strong>disabled</strong>.</td>
</tr>
<tr>
<td>Local-Advertisement</td>
<td>Status of advertisements that the local interface sends to the peer.</td>
</tr>
<tr>
<td>Enable</td>
<td>(DCBX Version 1.01 only)</td>
</tr>
<tr>
<td></td>
<td>State that the local interface advertises to the peer:</td>
</tr>
<tr>
<td></td>
<td>• <strong>Yes</strong>—The feature is enabled.</td>
</tr>
<tr>
<td></td>
<td>• <strong>No</strong>—The feature is disabled.</td>
</tr>
<tr>
<td>Willing</td>
<td>Willingness of the local interface to learn the PFC configuration from the peer</td>
</tr>
<tr>
<td></td>
<td>using DCBX:</td>
</tr>
<tr>
<td></td>
<td>• <strong>Yes</strong>—The local interface is willing to learn the PFC configuration from the</td>
</tr>
<tr>
<td></td>
<td>peer.</td>
</tr>
<tr>
<td></td>
<td>• <strong>No</strong>—The local interface is not willing to learn the PFC configuration from the</td>
</tr>
<tr>
<td></td>
<td>peer.</td>
</tr>
<tr>
<td>Mac auth Bypass Capability</td>
<td>(IEEE DCBX only)</td>
</tr>
<tr>
<td></td>
<td>(QFX Series) Media access controller (MAC) authentication bypass provides access</td>
</tr>
<tr>
<td></td>
<td>to devices based on MAC address authentication. This is not supported, so the only</td>
</tr>
<tr>
<td></td>
<td>value seen in the local advertisement field is <strong>no</strong>.</td>
</tr>
<tr>
<td>Error</td>
<td>(DCBX Version 1.01 only)</td>
</tr>
<tr>
<td></td>
<td>Configuration compatibility error status:</td>
</tr>
<tr>
<td></td>
<td>• <strong>No</strong>—No error detected. Local and peer configuration are compatible.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Yes</strong>—Error detected. Local and peer configuration are not compatible.</td>
</tr>
</tbody>
</table>
Table 20: show dcbx neighbors Output Fields *(continued)*

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Operational State</strong></td>
<td>PFC operational state on the interface:</td>
</tr>
<tr>
<td></td>
<td>• Enabled—PFC is enabled on the interface</td>
</tr>
<tr>
<td></td>
<td>• Disabled—PFC is disabled on the interface</td>
</tr>
<tr>
<td><strong>Maximum Traffic Classes capable to support PFC</strong></td>
<td>Largest number of traffic classes the local interface supports for PFC:</td>
</tr>
<tr>
<td></td>
<td>• 6 (EX Series switches)</td>
</tr>
<tr>
<td></td>
<td>• 8 (QFX Series)</td>
</tr>
<tr>
<td><strong>Code Point</strong></td>
<td>PFC code point, which is specified in the 3-bit class-of-service field in the VLAN header.</td>
</tr>
<tr>
<td><strong>Admin Mode</strong></td>
<td>PFC administrative state for each code point on the local interface:</td>
</tr>
<tr>
<td></td>
<td>• Enabled—PFC is enabled for the code point.</td>
</tr>
<tr>
<td></td>
<td>• Disabled—PFC is disabled for the code point.</td>
</tr>
<tr>
<td><strong>Operational Mode</strong></td>
<td>(QFX Series) PFC operational mode for each code point:</td>
</tr>
<tr>
<td></td>
<td>• Enable—PFC is enabled on the code point.</td>
</tr>
<tr>
<td></td>
<td>• Disable—PFC is disabled on the code point.</td>
</tr>
<tr>
<td><strong>Peer-Advertisement</strong></td>
<td>Status of advertisements that the peer sends to the local interface.</td>
</tr>
<tr>
<td><strong>Enable</strong></td>
<td>(DCBX Version 1.01 only)</td>
</tr>
<tr>
<td></td>
<td>State that the peer advertises to the local interface:</td>
</tr>
<tr>
<td></td>
<td>• Yes—The feature is enabled.</td>
</tr>
<tr>
<td></td>
<td>• No—The feature is disabled.</td>
</tr>
<tr>
<td><strong>Willing</strong></td>
<td>Willingness of the peer to learn the PFC configuration from the local interface using DCBX:</td>
</tr>
<tr>
<td></td>
<td>• Yes—The peer is willing to learn the PFC configuration from the local interface.</td>
</tr>
<tr>
<td></td>
<td>• No—The peer is not willing to learn the PFC configuration from the local interface.</td>
</tr>
<tr>
<td><strong>Error</strong></td>
<td>(DCBX Version 1.01 only)</td>
</tr>
<tr>
<td></td>
<td>Configuration compatibility error status:</td>
</tr>
<tr>
<td></td>
<td>• No—No error detected. Local and peer configuration are compatible.</td>
</tr>
<tr>
<td></td>
<td>• Yes—Error detected. Local and peer configuration are not compatible.</td>
</tr>
</tbody>
</table>
Table 20: show dcbx neighbors Output Fields (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Operational State</strong></td>
<td>PFC operational state on the interface:</td>
</tr>
<tr>
<td></td>
<td>• Enabled—PFC is enabled on the interface</td>
</tr>
<tr>
<td></td>
<td>• Disabled—PFC is disabled on the interface</td>
</tr>
<tr>
<td><strong>Mac auth Bypass Capability</strong></td>
<td>(IEEE DCBX only)</td>
</tr>
<tr>
<td></td>
<td>(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:</td>
</tr>
<tr>
<td></td>
<td>• Yes—The connected peer supports MAC authentication bypass.</td>
</tr>
<tr>
<td></td>
<td>• No—The connected peer does not support MAC authentication bypass.</td>
</tr>
<tr>
<td><strong>Maximum Traffic Classes</strong></td>
<td>Largest number of traffic classes the peer supports for PFC:</td>
</tr>
<tr>
<td><strong>capable to support PFC</strong></td>
<td>• 6 (EX Series switches)</td>
</tr>
<tr>
<td></td>
<td>• 8 (QFX Series)</td>
</tr>
<tr>
<td><strong>Code Point</strong></td>
<td>PFC code point, which is specified in the 3-bit class-of-service field in the VLAN header.</td>
</tr>
<tr>
<td><strong>Admin Mode</strong></td>
<td>PFC administrative state for each code point on the peer:</td>
</tr>
<tr>
<td></td>
<td>• Enabled—PFC is enabled for the code point.</td>
</tr>
<tr>
<td></td>
<td>• Disabled—PFC is disabled for the code point.</td>
</tr>
</tbody>
</table>
Table 20: show dcbx neighbors Output Fields (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feature: Application</td>
<td>State information for the DCBX application.</td>
</tr>
<tr>
<td>Protocol-State</td>
<td>(DCBX Version 1.01 only)</td>
</tr>
<tr>
<td></td>
<td>DCBX protocol state synchronization status:</td>
</tr>
<tr>
<td></td>
<td>• in-sync—The local interface received an acknowledge</td>
</tr>
<tr>
<td></td>
<td>message from the peer to indicate that the peer</td>
</tr>
<tr>
<td></td>
<td>received an FCoE state change message sent by the</td>
</tr>
<tr>
<td></td>
<td>local interface.</td>
</tr>
<tr>
<td></td>
<td>• ack-pending—The local interface has not yet received</td>
</tr>
<tr>
<td></td>
<td>an acknowledge message from the peer to indicate that</td>
</tr>
<tr>
<td></td>
<td>the peer received an FCoE state change message sent by</td>
</tr>
<tr>
<td></td>
<td>the local interface.</td>
</tr>
<tr>
<td></td>
<td>• not-applicable—The local interface is set to</td>
</tr>
<tr>
<td></td>
<td>no-auto-negotiation (autonegotiation is disabled). If</td>
</tr>
<tr>
<td></td>
<td>the interface is associated with an FCoE forwarding</td>
</tr>
<tr>
<td></td>
<td>class, the interface advertises FCoE capability even</td>
</tr>
<tr>
<td></td>
<td>if the connected peer does not advertise FCoE</td>
</tr>
<tr>
<td>Local-Advertisement</td>
<td>Status of advertisements that the local interface sends to</td>
</tr>
<tr>
<td></td>
<td>the peer.</td>
</tr>
<tr>
<td></td>
<td>If the local interface is set to no-auto-negotiation</td>
</tr>
<tr>
<td></td>
<td>(autonegotiation is disabled), the local advertisement</td>
</tr>
<tr>
<td></td>
<td>portion of the output is not shown.</td>
</tr>
<tr>
<td>Enable</td>
<td>(DCBX Version 1.01 only)</td>
</tr>
<tr>
<td></td>
<td>State that the local interface advertises to the peer:</td>
</tr>
<tr>
<td></td>
<td>• Yes—The feature is enabled.</td>
</tr>
<tr>
<td></td>
<td>• No—The feature is disabled.</td>
</tr>
<tr>
<td>Willing</td>
<td>(DCBX Version 1.01 only)</td>
</tr>
<tr>
<td></td>
<td>Willingness of the local interface to learn the FCoE</td>
</tr>
<tr>
<td></td>
<td>interface state from the peer using DCBX:</td>
</tr>
<tr>
<td></td>
<td>• Yes—The local interface is willing to learn the FCoE</td>
</tr>
<tr>
<td></td>
<td>interface state from the peer.</td>
</tr>
<tr>
<td></td>
<td>• No—The local interface is not willing to learn the</td>
</tr>
<tr>
<td></td>
<td>FCoE interface state from the peer.</td>
</tr>
<tr>
<td>Error</td>
<td>(DCBX Version 1.01 only)</td>
</tr>
<tr>
<td></td>
<td>Configuration compatibility error status:</td>
</tr>
<tr>
<td></td>
<td>• No—No error detected. The local and peer configuration</td>
</tr>
<tr>
<td></td>
<td>are compatible.</td>
</tr>
<tr>
<td></td>
<td>• Yes—Error detected. The local and peer configuration</td>
</tr>
<tr>
<td></td>
<td>are not compatible.</td>
</tr>
<tr>
<td>Appl-Name</td>
<td>Name of the application:</td>
</tr>
</tbody>
</table>
### Table 20: show dcbx neighbors Output Fields (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Ethernet-Type</strong></td>
<td>(DCBX Version 1.01 only)</td>
</tr>
<tr>
<td></td>
<td>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.</td>
</tr>
<tr>
<td><strong>Socket-Number</strong></td>
<td>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.</td>
</tr>
<tr>
<td><strong>Priority-Field or Priority-Map</strong></td>
<td>Priority assigned to the application. 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.</td>
</tr>
<tr>
<td><strong>Status</strong></td>
<td>(DCBX Version 1.01 only)</td>
</tr>
<tr>
<td></td>
<td>Local status when autonegotiation is enabled:</td>
</tr>
<tr>
<td></td>
<td>• <strong>Enabled</strong>—The application feature is enabled on both the local interface and the peer interface. (The local configuration and the peer configuration match.)</td>
</tr>
<tr>
<td></td>
<td>• <strong>Disabled</strong>—The local configuration and the peer configuration do not match.</td>
</tr>
<tr>
<td><strong>Peer-Advertisement</strong></td>
<td>Status of advertisements that the peer sends to the local interface.</td>
</tr>
<tr>
<td><strong>Enable</strong></td>
<td>(DCBX Version 1.01 only)</td>
</tr>
<tr>
<td></td>
<td>State that the peer advertises to the local interface:</td>
</tr>
<tr>
<td></td>
<td>• <strong>Yes</strong>—The feature is enabled.</td>
</tr>
<tr>
<td></td>
<td>• <strong>No</strong>—The feature is disabled.</td>
</tr>
<tr>
<td><strong>Willing</strong></td>
<td>(DCBX Version 1.01 only)</td>
</tr>
<tr>
<td></td>
<td>Willingness of the peer to learn the FCoE interface state from the local interface using DCBX:</td>
</tr>
<tr>
<td></td>
<td>• <strong>Yes</strong>—The peer is willing to learn the FCoE interface state from the local interface.</td>
</tr>
<tr>
<td></td>
<td>• <strong>No</strong>—The peer is not willing to learn the FCoE interface state from the local interface.</td>
</tr>
</tbody>
</table>
Table 20: show dcbx neighbors Output Fields (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Error</td>
<td>(DCBX Version 1.01 only)</td>
</tr>
<tr>
<td></td>
<td>Configuration compatibility error status:</td>
</tr>
<tr>
<td></td>
<td>• No—No error detected. Local and peer configuration are compatible.</td>
</tr>
<tr>
<td></td>
<td>• Yes—Error detected. Local and peer configuration are not compatible.</td>
</tr>
<tr>
<td>Appl-Name</td>
<td>Name of the application:</td>
</tr>
<tr>
<td></td>
<td>• FCoE—Fibre Channel over Ethernet</td>
</tr>
<tr>
<td>Ethernet-Type</td>
<td>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.</td>
</tr>
<tr>
<td>Socket-Number</td>
<td>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.</td>
</tr>
<tr>
<td>Priority-Field or Priority-Map</td>
<td>Priority assigned to the application.</td>
</tr>
<tr>
<td>Status</td>
<td>(DCBX Version 1.01 only)</td>
</tr>
<tr>
<td></td>
<td>Peer interface status:</td>
</tr>
<tr>
<td></td>
<td>• Enabled—The application feature is enabled on both the local interface and the peer interface. (The local configuration and the peer configuration match.)</td>
</tr>
<tr>
<td></td>
<td>• Disabled—The local configuration and the peer configuration do not match.</td>
</tr>
</tbody>
</table>
Table 20: show dcbx neighbors Output Fields (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feature: ETS</td>
<td>Enhanced Transmission Selection (ETS) DCBX state information.</td>
</tr>
<tr>
<td>Protocol-State</td>
<td>(DCBX Version 1.01 only) ETS protocol state synchronization status:</td>
</tr>
<tr>
<td></td>
<td>• 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.</td>
</tr>
<tr>
<td></td>
<td>• 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.</td>
</tr>
<tr>
<td>Operational State</td>
<td>(DCBX Version 1.01 only) Operational state of the feature, enabled or disabled.</td>
</tr>
<tr>
<td>Local-Advertisement</td>
<td>Status of advertisements that the local interface sends to the peer.</td>
</tr>
<tr>
<td>Enable</td>
<td>(DCBX Version 1.01 only) State that the local interface advertises to the peer:</td>
</tr>
<tr>
<td></td>
<td>• Yes—The feature is enabled.</td>
</tr>
<tr>
<td></td>
<td>• No—The feature is disabled.</td>
</tr>
<tr>
<td>TLV Type</td>
<td>(IEEE DCBX only) Type of ETS TLV:</td>
</tr>
<tr>
<td></td>
<td>• Configuration—Advertises the Configuration TLV, which communicates the local ETS configuration to the peer but does not ask the peer to use the configuration.</td>
</tr>
<tr>
<td></td>
<td>• 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.</td>
</tr>
<tr>
<td></td>
<td>• Recommendation-or-Configuration—Advertises both TLVs.</td>
</tr>
<tr>
<td>Willing</td>
<td>Willingness of the local interface to learn the ETS state from the peer using DCBX (EX Series switches always advertise No for this field):</td>
</tr>
<tr>
<td></td>
<td>• Yes—Local interface is willing to learn the ETS state from the peer.</td>
</tr>
<tr>
<td></td>
<td>• No—Local interface is not willing to learn the ETS state from the peer.</td>
</tr>
<tr>
<td>Credit Based Shaper</td>
<td></td>
</tr>
</tbody>
</table>
Table 20: show dcbx neighbors Output Fields (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>(IEEE DCBX only)</td>
<td>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.</td>
</tr>
<tr>
<td>Error</td>
<td>(DCBX Version 1.01 only)</td>
</tr>
<tr>
<td>Configuration error status:</td>
<td>• No—No error. This should always be the switch ETS error state.</td>
</tr>
<tr>
<td>• Yes—Error detected.</td>
<td></td>
</tr>
<tr>
<td>Maximum Traffic Classes capable to support PFC</td>
<td>(DCBX Version 1.01 only)</td>
</tr>
<tr>
<td>Largest number of traffic classes the local interface supports for PFC.</td>
<td></td>
</tr>
<tr>
<td>Maximum Traffic Classes supported</td>
<td>(IEEE DCBX only)</td>
</tr>
<tr>
<td>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.)</td>
<td></td>
</tr>
<tr>
<td>Code Point</td>
<td>PFC code point, which is specified in the 3-bit class-of-service field in the VLAN header.</td>
</tr>
<tr>
<td>Priority-Group</td>
<td>Class-of-service (CoS) priority group (forwarding class set) identification number.</td>
</tr>
<tr>
<td>Percentage B/W</td>
<td>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.)</td>
</tr>
<tr>
<td>Transmission Selection Algorithm</td>
<td>(IEEE DCBX only)</td>
</tr>
<tr>
<td>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.</td>
<td></td>
</tr>
<tr>
<td>Peer-Advertisement</td>
<td>Status of advertisements that the peer sends to the local interface.</td>
</tr>
<tr>
<td>Enable</td>
<td></td>
</tr>
<tr>
<td>Field Name</td>
<td>Field Description</td>
</tr>
<tr>
<td>------------------------------------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>TLV Type</strong></td>
<td><strong>(IEEE DCBX only)</strong></td>
</tr>
<tr>
<td>Type of ETS TLV</td>
<td>Type of ETS TLV:</td>
</tr>
<tr>
<td>• Configuration—Advertises the Configuration TLV, which communicates the local ETS configuration to the peer but does not ask the peer to use the configuration.</td>
<td></td>
</tr>
<tr>
<td>• 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.</td>
<td></td>
</tr>
<tr>
<td>• Configuration/Recommendation—Advertises both TLVs.</td>
<td></td>
</tr>
<tr>
<td><strong>Willing</strong></td>
<td><strong>(IEEE DCBX only)</strong></td>
</tr>
<tr>
<td>Willingness of the peer to learn the ETS state from the local interface using DCBX:</td>
<td></td>
</tr>
<tr>
<td>• Yes—Peer is willing to learn the ETS state from the local interface.</td>
<td></td>
</tr>
<tr>
<td>• No—Peer is not willing to learn the ETS state from the local interface.</td>
<td></td>
</tr>
<tr>
<td><strong>Credit Based Shaper</strong></td>
<td><strong>(IEEE DCBX only)</strong></td>
</tr>
<tr>
<td>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 <strong>No</strong>.</td>
<td></td>
</tr>
<tr>
<td><strong>Error</strong></td>
<td><strong>(DCBX Version 1.01 only)</strong></td>
</tr>
<tr>
<td>Configuration error status of the peer:</td>
<td></td>
</tr>
<tr>
<td>• No—No error in peer ETS TLV.</td>
<td></td>
</tr>
<tr>
<td>• Yes—Error in peer ETS TLV.</td>
<td></td>
</tr>
<tr>
<td><strong>Maximum Traffic Classes capable to support PFC</strong></td>
<td><strong>(DCBX Version 1.01 only)</strong></td>
</tr>
<tr>
<td>Largest number of traffic classes the local interface supports for PFC.</td>
<td></td>
</tr>
<tr>
<td><strong>Maximum Traffic Classes supported</strong></td>
<td><strong>(IEEE DCBX only)</strong></td>
</tr>
<tr>
<td>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.)</td>
<td></td>
</tr>
<tr>
<td><strong>Code Point</strong></td>
<td></td>
</tr>
</tbody>
</table>
Table 20: show dcbx neighbors Output Fields (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PFC code point, which is specified in the 3-bit class-of-service field in the VLAN header.</td>
<td></td>
</tr>
<tr>
<td>Priority-Group</td>
<td>CoS priority group (forwarding class set) identification number.</td>
</tr>
<tr>
<td>Percentage B/W</td>
<td>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.)</td>
</tr>
<tr>
<td>Transmission Selection Algorithm</td>
<td>(IEEE DCBX only)</td>
</tr>
<tr>
<td></td>
<td>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.</td>
</tr>
<tr>
<td>PFC</td>
<td>(QFX Series, terse option only) DCBX TLV advertisement state for PFC:</td>
</tr>
<tr>
<td></td>
<td>• Disabled—PFC configuration matches the configuration on the connected peer and PFC is disabled</td>
</tr>
<tr>
<td></td>
<td>• Enabled—PFC configuration matches the configuration on the connected peer and PFC is enabled</td>
</tr>
<tr>
<td></td>
<td>• Not Advt—Interface does not advertise PFC to the connected peer</td>
</tr>
<tr>
<td>ETS</td>
<td>(terse option only) Local DCBX TLV advertisement state for ETS:</td>
</tr>
<tr>
<td></td>
<td>• Advt—Interface advertises ETS TLVs</td>
</tr>
<tr>
<td></td>
<td>• Disabled—ETS is disabled on the interface (interface does not advertise ETS)</td>
</tr>
<tr>
<td>ETS Rec</td>
<td>(terse option only) DCBX TLV peer advertisement state for ETS (state received from the connected DCBX peer):</td>
</tr>
<tr>
<td></td>
<td>• Advt—Peer interface advertises ETS TLVs</td>
</tr>
<tr>
<td></td>
<td>• Not Advt—Peer interface does not advertise ETS</td>
</tr>
</tbody>
</table>

NOTE: When the DCBX mode is DCBX version 1.01, no peer information is displayed.
### Table 20: show dcbx neighbors Output Fields (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
</tr>
</thead>
</table>
| Version    | **(terse option only)** The DCBX version used on the interface and whether the DCBX version was autonegotiated or explicitly configured:  
  - IEEE—The interface uses IEEE DCBX.  
  - 1.01—The interface uses DCBX version 1.01.  
  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. |

### 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

<table>
<thead>
<tr>
<th>Code Point</th>
<th>Admin Mode</th>
<th>Operational Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>Disabled</td>
<td>Disable</td>
</tr>
<tr>
<td>001</td>
<td>Disabled</td>
<td>Disable</td>
</tr>
<tr>
<td>010</td>
<td>Disabled</td>
<td>Disable</td>
</tr>
<tr>
<td>011</td>
<td>Enabled</td>
<td>Enable</td>
</tr>
<tr>
<td>100</td>
<td>Enabled</td>
<td>Enable</td>
</tr>
<tr>
<td>101</td>
<td>Disabled</td>
<td>Disable</td>
</tr>
<tr>
<td>110</td>
<td>Disabled</td>
<td>Disable</td>
</tr>
<tr>
<td>111</td>
<td>Disabled</td>
<td>Disable</td>
</tr>
</tbody>
</table>

Peer-Advertisement:
  Enable: Yes, Willing: No, Error: No
  Maximum Traffic Classes capable to support PFC: 8

<table>
<thead>
<tr>
<th>Code Point</th>
<th>Admin Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>Disabled</td>
</tr>
</tbody>
</table>
```
Feature: Application, Protocol-State: in-sync

Local-Advertisement:
Enable: Yes, Willing: No, Error: No

<table>
<thead>
<tr>
<th>Appl-Name</th>
<th>Ethernet-Type</th>
<th>Socket-Number</th>
<th>Priority-Map</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>FCoE</td>
<td>0x8906</td>
<td></td>
<td>00001110</td>
<td>Enabled</td>
</tr>
<tr>
<td>iSCSI</td>
<td></td>
<td>3260</td>
<td>10000000</td>
<td>Enabled</td>
</tr>
</tbody>
</table>

Peer-Advertisement:
Enable: Yes, Willing: Yes, Error: No

<table>
<thead>
<tr>
<th>Appl-Name</th>
<th>Ethernet-Type</th>
<th>Socket-Number</th>
<th>Priority-Map</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>FCoE</td>
<td>0x8906</td>
<td>N/A</td>
<td>00001110</td>
<td>Enabled</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Code Point</th>
<th>Priority-Group</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>0</td>
</tr>
<tr>
<td>001</td>
<td>7</td>
</tr>
<tr>
<td>010</td>
<td>7</td>
</tr>
<tr>
<td>011</td>
<td>7</td>
</tr>
<tr>
<td>100</td>
<td>0</td>
</tr>
<tr>
<td>101</td>
<td>1</td>
</tr>
<tr>
<td>110</td>
<td>1</td>
</tr>
<tr>
<td>111</td>
<td>7</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Priority-Group</th>
<th>Percentage B/W</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>40%</td>
</tr>
<tr>
<td>1</td>
<td>5%</td>
</tr>
</tbody>
</table>

Peer-Advertisement:
Enable: Yes, Willing: No, Error: No

Maximum Traffic Classes capable to support PFC: 8

<table>
<thead>
<tr>
<th>Code Point</th>
<th>Priority-Group</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>0</td>
</tr>
<tr>
<td>001</td>
<td>7</td>
</tr>
<tr>
<td>010</td>
<td>7</td>
</tr>
<tr>
<td>011</td>
<td>7</td>
</tr>
<tr>
<td>100</td>
<td>0</td>
</tr>
<tr>
<td>101</td>
<td>1</td>
</tr>
<tr>
<td>110</td>
<td>1</td>
</tr>
</tbody>
</table>
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

<table>
<thead>
<tr>
<th>Code Point</th>
<th>Admin Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>Disabled</td>
</tr>
<tr>
<td>001</td>
<td>Disabled</td>
</tr>
<tr>
<td>010</td>
<td>Disabled</td>
</tr>
<tr>
<td>011</td>
<td>Enabled</td>
</tr>
<tr>
<td>100</td>
<td>Enabled</td>
</tr>
<tr>
<td>101</td>
<td>Disabled</td>
</tr>
<tr>
<td>110</td>
<td>Disabled</td>
</tr>
<tr>
<td>111</td>
<td>Disabled</td>
</tr>
</tbody>
</table>

Peer-Advertisement:
Willing: No
Mac auth Bypass Capability: No
Operational State: Enabled

Maximum Traffic Classes capable to support PFC: 8

<table>
<thead>
<tr>
<th>Code Point</th>
<th>Admin Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>Disabled</td>
</tr>
<tr>
<td>001</td>
<td>Disabled</td>
</tr>
<tr>
<td>010</td>
<td>Disabled</td>
</tr>
<tr>
<td>011</td>
<td>Enabled</td>
</tr>
<tr>
<td>100</td>
<td>Enabled</td>
</tr>
<tr>
<td>101</td>
<td>Disabled</td>
</tr>
<tr>
<td>110</td>
<td>Disabled</td>
</tr>
<tr>
<td>111</td>
<td>Disabled</td>
</tr>
</tbody>
</table>

Feature: Application

Local-Advertisement:

<table>
<thead>
<tr>
<th>Appl-Name</th>
<th>Ethernet-Type</th>
<th>Socket-Number</th>
<th>Priority-field</th>
</tr>
</thead>
<tbody>
<tr>
<td>FCoE</td>
<td>0x8906</td>
<td></td>
<td>00001110</td>
</tr>
<tr>
<td>iSCSI</td>
<td>3260</td>
<td></td>
<td>10000000</td>
</tr>
</tbody>
</table>

Peer-Advertisement:

<table>
<thead>
<tr>
<th>Appl-Name</th>
<th>Ethernet-Type</th>
<th>Socket-Number</th>
<th>Priority-field</th>
</tr>
</thead>
</table>
**Feature: ETS**

**Local-Advertisement:**
- **TLV Type:** Configuration/Recommendation
- **Willing:** No
- **Credit Based Shaper:** No
- **Maximum Traffic Classes supported:** 3

<table>
<thead>
<tr>
<th>Code Point</th>
<th>Priority-Group</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>0</td>
</tr>
<tr>
<td>001</td>
<td>7</td>
</tr>
<tr>
<td>010</td>
<td>7</td>
</tr>
<tr>
<td>011</td>
<td>7</td>
</tr>
<tr>
<td>100</td>
<td>0</td>
</tr>
<tr>
<td>101</td>
<td>1</td>
</tr>
<tr>
<td>110</td>
<td>1</td>
</tr>
<tr>
<td>111</td>
<td>7</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Priority-Group</th>
<th>Percentage B/W</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>40%</td>
</tr>
<tr>
<td>1</td>
<td>5%</td>
</tr>
</tbody>
</table>

**Transmission Selection Algorithm**
- 0: Enhanced Transmission Selection
- 1: Enhanced Transmission Selection

**Peer-Advertisement:**
- **TLV Type:** Configuration
- **Willing:** No
- **Credit Based Shaper:** No

<table>
<thead>
<tr>
<th>Code Point</th>
<th>Priority-Group</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>0</td>
</tr>
<tr>
<td>001</td>
<td>7</td>
</tr>
<tr>
<td>010</td>
<td>7</td>
</tr>
<tr>
<td>011</td>
<td>7</td>
</tr>
<tr>
<td>100</td>
<td>0</td>
</tr>
<tr>
<td>101</td>
<td>1</td>
</tr>
<tr>
<td>110</td>
<td>1</td>
</tr>
<tr>
<td>111</td>
<td>7</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Priority-Group</th>
<th>Percentage B/W</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>40%</td>
</tr>
<tr>
<td>1</td>
<td>5%</td>
</tr>
</tbody>
</table>

**Transmission Selection Algorithm**
- 0: Enhanced Transmission Selection
- 1: Enhanced Transmission Selection

**Peer-Advertisement:**
- **TLV Type:** Recommendation

<table>
<thead>
<tr>
<th>Code Point</th>
<th>Priority-Group</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>0</td>
</tr>
<tr>
<td>001</td>
<td>7</td>
</tr>
<tr>
<td>010</td>
<td>7</td>
</tr>
<tr>
<td>011</td>
<td>7</td>
</tr>
<tr>
<td>100</td>
<td>0</td>
</tr>
</tbody>
</table>

**Copyright © 2014, Juniper Networks, Inc.**
show dcbx neighbors terse (QFX Series)

user@switch> show dcbx neighbors terse

<table>
<thead>
<tr>
<th>Interface</th>
<th>Parent</th>
<th>PFC</th>
<th>ETS</th>
<th>ETS</th>
<th>Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>xe-0/0/8.0</td>
<td>-</td>
<td>Enabled</td>
<td>Advt</td>
<td>Advt</td>
<td>IEEE (Auto)</td>
</tr>
<tr>
<td>xe-0/0/9.0</td>
<td>-</td>
<td>Disabled</td>
<td>Disabled</td>
<td></td>
<td>1.01</td>
</tr>
<tr>
<td>xe-0/0/11.0 ae0.0</td>
<td>enabled</td>
<td>Advt</td>
<td>Advt</td>
<td></td>
<td>IEEE (Auto)</td>
</tr>
<tr>
<td>xe-0/0/12.0 ae0.0</td>
<td>enabled</td>
<td>Advt</td>
<td>Advt</td>
<td></td>
<td>IEEE (Auto)</td>
</tr>
<tr>
<td>xe-0/0/32.0</td>
<td>-</td>
<td>Enabled</td>
<td>Advt</td>
<td>Not Advt</td>
<td>IEEE</td>
</tr>
<tr>
<td>xe-0/0/36.0</td>
<td>-</td>
<td>Not Advt</td>
<td>Advt</td>
<td>Advt</td>
<td>IEEE</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Code Point</th>
<th>Admin Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>Disabled</td>
</tr>
<tr>
<td>001</td>
<td>Disabled</td>
</tr>
<tr>
<td>010</td>
<td>Disabled</td>
</tr>
<tr>
<td>011</td>
<td>Enabled</td>
</tr>
<tr>
<td>100</td>
<td>Disabled</td>
</tr>
<tr>
<td>101</td>
<td>Disabled</td>
</tr>
<tr>
<td>110</td>
<td>Disabled</td>
</tr>
<tr>
<td>111</td>
<td>Disabled</td>
</tr>
</tbody>
</table>
Peer-Advertisement:
Enable: Yes, Willing: No, Error: No
Maximum Traffic Classes capable to support PFC: 6

<table>
<thead>
<tr>
<th>Code Point</th>
<th>Admin Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>Disabled</td>
</tr>
<tr>
<td>001</td>
<td>Disabled</td>
</tr>
<tr>
<td>010</td>
<td>Disabled</td>
</tr>
<tr>
<td>011</td>
<td>Enabled</td>
</tr>
<tr>
<td>100</td>
<td>Disabled</td>
</tr>
<tr>
<td>101</td>
<td>Disabled</td>
</tr>
<tr>
<td>110</td>
<td>Disabled</td>
</tr>
<tr>
<td>111</td>
<td>Disabled</td>
</tr>
</tbody>
</table>

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.

<table>
<thead>
<tr>
<th>Appl-Name</th>
<th>Ethernet-Type</th>
<th>Socket-Number</th>
<th>Priority-Map</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>FCoE</td>
<td>0x8906</td>
<td></td>
<td>00001000</td>
<td>Enabled</td>
</tr>
</tbody>
</table>

Peer-Advertisement:
Enable: Yes, Willing: No, Error: No

<table>
<thead>
<tr>
<th>Status</th>
<th>Appl-Name</th>
<th>Ethernet-Type</th>
<th>Socket-Number</th>
<th>Priority-Map</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enabled</td>
<td>FCoE</td>
<td>0x8906</td>
<td></td>
<td>00001000</td>
</tr>
</tbody>
</table>

**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

<table>
<thead>
<tr>
<th>Code Point</th>
<th>Admin Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>Disabled</td>
</tr>
<tr>
<td>001</td>
<td>Disabled</td>
</tr>
<tr>
<td>010</td>
<td>Disabled</td>
</tr>
<tr>
<td>011</td>
<td>Enabled</td>
</tr>
<tr>
<td>100</td>
<td>Disabled</td>
</tr>
<tr>
<td>101</td>
<td>Disabled</td>
</tr>
<tr>
<td>110</td>
<td>Disabled</td>
</tr>
<tr>
<td>111</td>
<td>Disabled</td>
</tr>
</tbody>
</table>

Peer-Advertisement:
Enable: Yes, Willing: No, Error: No
Maximum Traffic Classes capable to support PFC: 6

<table>
<thead>
<tr>
<th>Code Point</th>
<th>Admin Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>Disabled</td>
</tr>
<tr>
<td>001</td>
<td>Disabled</td>
</tr>
<tr>
<td>010</td>
<td>Disabled</td>
</tr>
<tr>
<td>011</td>
<td>Enabled</td>
</tr>
<tr>
<td>100</td>
<td>Disabled</td>
</tr>
<tr>
<td>101</td>
<td>Disabled</td>
</tr>
<tr>
<td>110</td>
<td>Disabled</td>
</tr>
<tr>
<td>111</td>
<td>Disabled</td>
</tr>
</tbody>
</table>

Feature: Application, Protocol-State: in-sync
Local-Advertisement:
Enable: Yes, Willing: No, Error: No

<table>
<thead>
<tr>
<th>Appl-Name</th>
<th>Ethernet-Type</th>
<th>Socket-Number</th>
<th>Priority-Map</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>FCoE</td>
<td>0x8906</td>
<td></td>
<td>00001000</td>
<td>Enabled</td>
</tr>
<tr>
<td>iscsi</td>
<td></td>
<td>3260</td>
<td>00100000</td>
<td>Enabled</td>
</tr>
</tbody>
</table>

Peer-Advertisement:
Enable: Yes, Willing: No, Error: No

<table>
<thead>
<tr>
<th>Appl-Name</th>
<th>Ethernet-Type</th>
<th>Socket-Number</th>
<th>Priority-Map</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>FCoE</td>
<td>0x8906</td>
<td></td>
<td>00001000</td>
<td>Enabled</td>
</tr>
<tr>
<td>iscsi</td>
<td></td>
<td>3260</td>
<td>00100000</td>
<td>Enabled</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Code Point</th>
<th>Admin Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>Enabled</td>
</tr>
<tr>
<td>001</td>
<td>Enabled</td>
</tr>
<tr>
<td>010</td>
<td>Disabled</td>
</tr>
<tr>
<td>011</td>
<td>Disabled</td>
</tr>
<tr>
<td>100</td>
<td>Disabled</td>
</tr>
<tr>
<td>101</td>
<td>Disabled</td>
</tr>
<tr>
<td>110</td>
<td>Disabled</td>
</tr>
<tr>
<td>111</td>
<td>Disabled</td>
</tr>
</tbody>
</table>

Peer-Advertisement:
  Enable: Yes, Willing: Yes, Error: No
  Maximum Traffic Classes capable to support PFC: 8

<table>
<thead>
<tr>
<th>Code Point</th>
<th>Admin Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>Enabled</td>
</tr>
<tr>
<td>001</td>
<td>Disabled</td>
</tr>
<tr>
<td>010</td>
<td>Disabled</td>
</tr>
<tr>
<td>011</td>
<td>Disabled</td>
</tr>
<tr>
<td>100</td>
<td>Enabled</td>
</tr>
<tr>
<td>101</td>
<td>Disabled</td>
</tr>
<tr>
<td>110</td>
<td>Disabled</td>
</tr>
<tr>
<td>111</td>
<td>Disabled</td>
</tr>
</tbody>
</table>

Feature: Application, Protocol-State: in-sync

Local-Advertisement:
  Enable: Yes, Willing: No, Error: No

<table>
<thead>
<tr>
<th>Appl-Name</th>
<th>Ethernet-Type</th>
<th>Socket-Number</th>
<th>Priority-Map</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>FCoE</td>
<td>0x8906</td>
<td>3260</td>
<td>00000010</td>
<td>Enabled</td>
</tr>
<tr>
<td>iscsi</td>
<td>3260</td>
<td>00000010</td>
<td>Enabled</td>
<td></td>
</tr>
</tbody>
</table>

Peer-Advertisement:
  Enable: Yes, Willing: Yes, Error: No

<table>
<thead>
<tr>
<th>Appl-Name</th>
<th>Ethernet-Type</th>
<th>Socket-Number</th>
<th>Priority-Map</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>FCoE</td>
<td>0x8906</td>
<td>3260</td>
<td>00000010</td>
<td>Enabled</td>
</tr>
<tr>
<td>iscsi</td>
<td>3260</td>
<td>00000010</td>
<td>Enabled</td>
<td></td>
</tr>
</tbody>
</table>

Feature: ETS, Protocol-State: in-sync

Operational State: Enabled
Local Advertisement:
Enable: Yes, Willing: No, Error: No
Maximum Traffic Classes supported: 3

<table>
<thead>
<tr>
<th>Code Point</th>
<th>Priority-Group</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>7</td>
</tr>
<tr>
<td>001</td>
<td>7</td>
</tr>
<tr>
<td>010</td>
<td>7</td>
</tr>
<tr>
<td>011</td>
<td>7</td>
</tr>
<tr>
<td>100</td>
<td>7</td>
</tr>
<tr>
<td>101</td>
<td>7</td>
</tr>
<tr>
<td>110</td>
<td>7</td>
</tr>
<tr>
<td>111</td>
<td>7</td>
</tr>
</tbody>
</table>

Priority-Group | Percentage B/W
----------------|------------------
7               | 100%             

Peer Advertisement:
Enable: Yes, Willing: Yes, Error: No
Maximum Traffic Classes supported: 8

<table>
<thead>
<tr>
<th>Code Point</th>
<th>Priority-Group</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>0</td>
</tr>
<tr>
<td>001</td>
<td>1</td>
</tr>
<tr>
<td>010</td>
<td>0</td>
</tr>
<tr>
<td>011</td>
<td>0</td>
</tr>
<tr>
<td>100</td>
<td>2</td>
</tr>
<tr>
<td>101</td>
<td>0</td>
</tr>
<tr>
<td>110</td>
<td>0</td>
</tr>
<tr>
<td>111</td>
<td>0</td>
</tr>
</tbody>
</table>

Priority-Group | Percentage B/W
----------------|------------------
0               | 30%              |
1               | 40%              |
2               | 30%              |
show fibre-channel fabric

Syntax

show fibre-channel fabric
  <extensive | summary>
  <fabric-name>
  <sort-by (name | fabric-id)>

Release Information

Command introduced in Junos OS Release 11.1 for the QFX Series.

Description

Display Fibre Channel fabric information.

Options

fabric-name—(Optional) Display output only for the specified fabric.

extensive | summary—(Optional) Display the specified level of output.

sort-by (name | fabric-id)—(Optional) Sort output by fabric name or fabric ID.

Required Privilege

view

Related Documentation

• fc-fabrics

• Configuring an FCoE-FC Gateway Fibre Channel Fabric

List of Sample Output

show fibre-channel fabric on page 272
show fibre-channel fabric extensive on page 272

Output Fields

Table 21 on page 271 lists the output fields for the show fibre-channel fabric command. Output fields are listed in the approximate order in which they appear.

Table 21: show fibre-channel fabric Output Fields

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
<th>Level of Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fabric</td>
<td>Name of the fabric.</td>
<td>All</td>
</tr>
<tr>
<td>Fabric-ID</td>
<td>Identification number of the fabric.</td>
<td>All</td>
</tr>
<tr>
<td>Type</td>
<td>Type of fabric. All fabrics are PROXY fabrics.</td>
<td>All</td>
</tr>
<tr>
<td>Interfaces</td>
<td>Native Fibre Channel interfaces and FCoE interfaces assigned to the fabric.</td>
<td>All</td>
</tr>
<tr>
<td>Created at</td>
<td>Date and time the fabric was created.</td>
<td>extensive</td>
</tr>
<tr>
<td>Internal Index</td>
<td>Fabric index internal to Junos OS.</td>
<td>extensive</td>
</tr>
<tr>
<td>Origin</td>
<td>Origin information internal to Junos OS.</td>
<td>extensive</td>
</tr>
<tr>
<td>Description</td>
<td>Text description of the fabric.</td>
<td>extensive</td>
</tr>
</tbody>
</table>
### Table 21: show fibre-channel fabric Output Fields (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
<th>Level of Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fabric WWN</td>
<td>Unique WWN of the fabric generated by the FCF.</td>
<td>extensive</td>
</tr>
<tr>
<td>Login sessions</td>
<td>Number of FIP login sessions currently running on the fabric.</td>
<td>extensive</td>
</tr>
<tr>
<td>Configured max login sessions</td>
<td>Configured maximum number of FIP login sessions permitted on the fabric.</td>
<td>extensive</td>
</tr>
</tbody>
</table>

### Sample Output

**show fibre-channel fabric**

```
user@switch> show fibre-channel fabric
Fabric         Fabric-ID     Type     Interfaces
proxy2         200           PROXY     fc-0/0/0.0
                    fc-0/0/1.0
```

**show fibre-channel fabric extensive**

```
user@switch> show fibre-channel fabric extensive
Fabric: proxy2, Created at: Mon Apr 19 14:02:58 2010
Fabric-ID: 200, Internal index: 2, Origin: Static
Description: srv-fabric, Type: PROXY, Fabric WWN: 10:00:00:05:33:51:d7:cd
Login sessions: 200, Configured max login sessions: 500
   fc-0/0/0.0, (untagged)
   fc-0/0/1.0, (untagged)
```
show fibre-channel fc2 sessions

Syntax

show fibre-channel fc2 sessions
<fabric fabric-name>
<brief | detail>

Release Information

Command introduced in Junos OS Release 11.1 for the QFX Series.

Description

Display Fibre Channel FC-2 information.

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

Options

brief | detail—(Optional) Display the specified level of output.

Required Privilege

view

Related Documentation

- show fibre-channel fc2 statistics on page 275
- clear fibre-channel fc2 statistics on page 227

List of Sample Output

show fibre-channel fc2 sessions on page 274
show fibre-channel fc2 sessions detail on page 274

Output Fields

Table 22 on page 273 lists the output fields for the show fibre-channel fc2 sessions command. Output fields are listed in the approximate order in which they appear.

Table 22: show fibre-channel fc2 sessions Output Fields

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
<th>Level of Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fabric</td>
<td>Name of the fabric.</td>
<td>All</td>
</tr>
<tr>
<td>Fabric-id</td>
<td>Identification number of the fabric.</td>
<td>All</td>
</tr>
<tr>
<td>Interface Name</td>
<td>Name of the interface.</td>
<td>All</td>
</tr>
<tr>
<td>Local FCID</td>
<td>Address of the local end of the connection.</td>
<td>All</td>
</tr>
<tr>
<td>Far FCID</td>
<td>Address of the far (remote) end of the connection.</td>
<td>All</td>
</tr>
<tr>
<td># Pending Exchanges</td>
<td>Number of pending exchanges for the session.</td>
<td>All</td>
</tr>
<tr>
<td>Flags</td>
<td>Flags internal to Junos OS.</td>
<td>detail</td>
</tr>
</tbody>
</table>
Table 22: show fibre-channel fc2 sessions Output Fields (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
<th>Level of Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Refcount</td>
<td>Reference count internal to Junos OS.</td>
<td>detail</td>
</tr>
<tr>
<td>Users</td>
<td>Information internal to Junos OS.</td>
<td>detail</td>
</tr>
</tbody>
</table>

Sample Output
show fibre-channel fc2 sessions

```
user@switch> show fibre-channel fc2 sessions
Fabric: fip-proxy, Fabric-id: 1
Interface Name      Local FCID:         Far FCID:           Exchanges
fc-0/0/0.0           *         0xfffffe    0
fc-0/0/1.0           *         0xfffffe    0
fc-0/0/2.0           *         0xfffffe    0
```

show fibre-channel fc2 sessions detail

```
user@switch> show fibre-channel fc2 sessions detail
Fabric: fip-proxy, Fabric-id: 1
Interface Name      Local FCID:         Far FCID:           Exchanges
fc-0/0/0.0           *         0xfffffe    0
   Flags:             SELF_LOCK USER_SYNCHED
   Refcount:          2
   Users:             1

Interface Name      Local FCID:         Far FCID:           Exchanges
fc-0/0/1.0           *         0xfffffe    0
   Flags:             SELF_LOCK USER_SYNCHED
   Refcount:          2

Interface Name      Local FCID:         Far FCID:           Exchanges
fc-0/0/2.0           *         0xfffffe    0
   Flags:             SELF_LOCK USER_SYNCHED
   Refcount:          2
   Users:             1
```
show fibre-channel fc2 statistics

Syntax
show fibre-channel fc2 statistics
<fabric fabric-name>

Release Information
Command introduced in Junos OS Release 11.1 for the QFX Series.

Description
Display Fibre Channel FC-2 statistics.

Options

Required Privilege Level
view

Related Documentation
• show fibre-channel fc2 sessions on page 273
• clear fibre-channel fc2 statistics on page 227

List of Sample Output
show fibre-channel fc2 statistics on page 276

Output Fields
Table 23 on page 275 lists the output fields for the show fibre-channel fc2 statistics command. Output fields are listed in the approximate order in which they appear.

Table 23: show fibre-channel fc2 statistics Output Fields

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Global statistics</td>
<td>Statistics for all fabrics.</td>
</tr>
<tr>
<td>Frame buffers allocated</td>
<td>Number of frame buffers currently allocated to all fabrics.</td>
</tr>
<tr>
<td>Frame buffers freed</td>
<td>Number of frame buffers freed.</td>
</tr>
<tr>
<td>Frames dropped</td>
<td>Number of dropped frames.</td>
</tr>
<tr>
<td>Fabric statistics</td>
<td>Fabric-specific statistics.</td>
</tr>
<tr>
<td>Fabric</td>
<td>Name of the fabric.</td>
</tr>
<tr>
<td>Fabric-id</td>
<td>Identification number of the fabric.</td>
</tr>
<tr>
<td>Tx-FRJTs</td>
<td>Number of fabric frame rejects (F_RJTs).</td>
</tr>
<tr>
<td>Tx-PRJTs</td>
<td>Number of port frame rejects (P_RJTs).</td>
</tr>
<tr>
<td>Tx-LSRJTs</td>
<td>Number of link service rejections.</td>
</tr>
<tr>
<td>Tx-ABTS</td>
<td>Number of abort sequence frames sent.</td>
</tr>
<tr>
<td>Rx-Drops</td>
<td>Number of received frames dropped.</td>
</tr>
</tbody>
</table>
Sample Output

show fibre-channel fc2 statistics

    user@switch> show fibre-channel fc2 statistics
    Global statistics:
    Frame buffers allocated: 60
    Frame buffers freed: 60
    Frames dropped: 0

    Fabric statistics:
    Fabric : fip-proxy, Fabric-id: 1
    Tx-FRJTs: 0
    Tx-PRJTs: 0
    Tx-LSRJTs: 0
    Tx-ABTS: 0
    Rx-Drops: 0
    Rx-ABTS: 0
show fibre-channel fip

Syntax

```
show fibre-channel fip
  <brief | detail>
```

Release Information

Command introduced in Junos OS Release 11.1 for the QFX Series.

Description

Display Fibre Channel over Ethernet Initialization Protocol (FIP) information.

Options

```
brief | detail — (Optional) Display the specified level of output.
```

Required Privilege Level

view

Related Documentation

- Configuring FIP on an FCoE-FC Gateway
- show fibre-channel fip enode on page 282
- show fibre-channel fip fabric on page 286
- show fibre-channel fip fcf on page 289
- show fibre-channel fip interface on page 292
- show fibre-channel fip statistics on page 295
- clear fibre-channel fip statistics on page 229

List of Sample Output

```
show fibre-channel fip on page 279
show fibre-channel fip detail on page 280
```

Output Fields

Table 24 on page 277 lists the output fields for the show fibre-channel fip command. Output fields are listed in the approximate order in which they appear. A session is a FLOGI or FDISC login to the FC SAN fabric. Session does not refer to end-to-end storage sessions.

Table 24: show fibre-channel fip Output Fields

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
<th>Level of Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configured max FIP sessions per Node Device</td>
<td>Configured maximum number of FIP sessions permitted on the Node device. For QFabric systems, this is the maximum number of FIP sessions permitted on each Node device in the fabric. For QFX3500 devices, this is the maximum number of FIP sessions permitted on the device.</td>
<td>detail</td>
</tr>
<tr>
<td>Node Device</td>
<td>Node device identifier.</td>
<td>detail</td>
</tr>
<tr>
<td>Total FIP sessions</td>
<td>Total number of FIP sessions on the FCoE-FC gateway switch.</td>
<td>detail</td>
</tr>
</tbody>
</table>
Table 24: show fibre-channel fip Output Fields (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
<th>Level of Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total FCoE filters</td>
<td>Total number of FIP filters on the FCoE-FC gateway switch.</td>
<td>detail</td>
</tr>
<tr>
<td>Fabric Name</td>
<td>Name of the fabric and in parentheses the fabric ID.</td>
<td>All</td>
</tr>
<tr>
<td>FCoE trusted</td>
<td>Whether ports on the FC fabric are trusted or untrusted:</td>
<td>detail</td>
</tr>
<tr>
<td></td>
<td>• Yes—Ports on the FC fabric are trusted; FIP snooping is turned off.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• No—Ports on the FC fabric are not trusted; FIP snooping is turned on.</td>
<td></td>
</tr>
<tr>
<td>FCoE trusted</td>
<td>Whether ports on the FC fabric are trusted or untrusted:</td>
<td>detail</td>
</tr>
<tr>
<td></td>
<td>• Yes—Ports on the FC fabric are trusted; FIP snooping is turned off.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• No—Ports on the FC fabric are not trusted; FIP snooping is turned on.</td>
<td></td>
</tr>
<tr>
<td>Member</td>
<td>Information about an FCF that is a member of the fabric.</td>
<td>All</td>
</tr>
<tr>
<td></td>
<td>• FCF-MAC MAC address used in discovery advertisements.</td>
<td>All</td>
</tr>
<tr>
<td></td>
<td>• FKA-ADV-PERIOD Period of time in milliseconds between FIP keepalive advertisements configured for the FC fabric.</td>
<td>detail</td>
</tr>
<tr>
<td></td>
<td>• FKA-ADV-D-BIT Disable FIP keepalive advertisement monitoring bit. The state is always off.</td>
<td>detail</td>
</tr>
<tr>
<td></td>
<td>• Type Type of interface:</td>
<td>detail</td>
</tr>
<tr>
<td></td>
<td>• VF_Port Capable—Interface can act as a VF_Port interface.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Priority Priority value associated with the switch FCF-MAC. Converged network adapters (CNAs) use the priority value to determine the switch with which they will perform FIP FLOGI. The lower the value, the higher the priority. Value range: 0 through 255.</td>
<td>detail</td>
</tr>
<tr>
<td></td>
<td>• State FIP state on the fabric:</td>
<td>detail</td>
</tr>
<tr>
<td></td>
<td>• Enable—FIP is enabled on the fabric.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Disable—FIP is disabled on the fabric.</td>
<td></td>
</tr>
</tbody>
</table>
### Table 24: show fibre-channel fip Output Fields (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
<th>Level of Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENode</td>
<td>Information about a connected FCoE node (ENode).</td>
<td>All</td>
</tr>
<tr>
<td></td>
<td>• ENode-MAC</td>
<td>All</td>
</tr>
<tr>
<td></td>
<td>MAC address of the connected ENode.</td>
<td>All</td>
</tr>
<tr>
<td></td>
<td>• Enode State</td>
<td>All</td>
</tr>
<tr>
<td></td>
<td>Login state internal to Junos OS.</td>
<td>All</td>
</tr>
<tr>
<td></td>
<td>• Configured ENode timer</td>
<td>detail</td>
</tr>
<tr>
<td></td>
<td>User-configured FIP keepalive advertisement interval in milliseconds.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Running ENode timer</td>
<td>detail</td>
</tr>
<tr>
<td></td>
<td>Runtime interval in milliseconds of the last FIP keepalive advertisement received.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Active FIP Sessions</td>
<td>detail</td>
</tr>
<tr>
<td></td>
<td>Number of active FIP sessions on the ENode.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• VN-Port-MAC</td>
<td>All</td>
</tr>
<tr>
<td></td>
<td>MAC address of a VN_Port on the ENode.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Session State</td>
<td>detail</td>
</tr>
<tr>
<td></td>
<td>Session state internal to Junos OS.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Configured FKA-ADV</td>
<td>detail</td>
</tr>
<tr>
<td></td>
<td>User-configured FIP keepalive advertisement interval in milliseconds.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Running FKA-ADV</td>
<td>detail</td>
</tr>
<tr>
<td></td>
<td>Runtime interval in milliseconds of the last FIP keepalive advertisement received.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Configured VN-Port Timer</td>
<td>detail</td>
</tr>
<tr>
<td></td>
<td>Configured state of the VN_Port keepalive timer in milliseconds. This value changes every time an FKA_ADV is received.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Running VN-Port Timer</td>
<td>detail</td>
</tr>
<tr>
<td></td>
<td>Running state of the VN_Port keepalive timer in milliseconds.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• FCID</td>
<td>detail</td>
</tr>
<tr>
<td></td>
<td>Fibre Channel ID of the VN_Port.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• WWN</td>
<td>detail</td>
</tr>
<tr>
<td></td>
<td>Unique worldwide name of the VN_Port.</td>
<td></td>
</tr>
</tbody>
</table>

### Sample Output

show fibre-channel fip

```
user@switch> show fibre-channel fip
Fabric Name : proxy2 (200)
  Member
  FCF-MAC : 00:30:48:b0:ee:d2 (Interface vlan.100)
  Enode
    Enode-MAC : 00:10:94:00:00:02     State : Logged-in
  Session
    VN-Port-MAC : 0e:fc:00:03:00:02
    VN-Port-MAC : 0e:fc:00:03:00:01
```
Enode-MAC : 00:10:94:00:00:03       State : Logged-in
Session
VN-Port-MAC              : 0e:fc:00:03:00:04
VN-Port-MAC              : 0e:fc:00:03:00:03

show fibre-channel fip detail

user@switch> show fibre-channel fip detail
Configured max FIP sessions per Node Device: 2500
Node Device: 0  Total FIP sessions: 4  Total FCoE filters: 4
Fabric Name : proxy2 (200)
FC-MAP         : 0e:fc:00
FKA-ADV-PERIOD : 90000        MAX-SESSIONS-PER-ENODE : 32
FCoE trusted   : No
Member
FCF-MAC: 00:30:48:b0:ee:d2  (Interface vlan.100)
FKA-ADV-PERIOD  : 90000    FKA-ADV-D-BIT-bit : Off
Type : VF_Port Capable
Priority : 86              State : Enable
ENode
Enode-MAC : 00:10:94:00:00:02   ENode State : Logged-in
Configured ENode timer: 8000    Running ENode timer: 12226
Active FIP Sessions : 2

Session details
VN-Port-MAC              : 0e:fc:00:03:00:02
Session state            : Up
Configured FKA-ADV       : 90000
Running FKA-ADV          : 0
Configured VN-Port Timer : 90000
Running VN-Port Timer    : 213193
FCID                     : 0x2c1a01
WWN                      : 10:00:00:00:c9:a4:a3:cf

VN-Port-MAC              : 0e:fc:00:03:00:01
Session state            : Up
Configured FKA-ADV       : 90000
Running FKA-ADV          : 0
Configured VN-Port Timer : 90000
Running VN-Port Timer    : 213632
FCID                     : 0x2c1a02
WWN                      : 10:00:00:00:d9:b4:e3:df

ENode
Enode-MAC : 00:10:94:00:00:03   ENode State : Logged-in
Configured ENode timer: 8000    Running ENode timer: 12254
Active FIP Sessions : 2

Session details
VN-Port-MAC              : 0e:fc:00:03:00:00
Session state            : Up
Configured FKA-ADV       : 90000
Running FKA-ADV          : 0
Configured VN-Port Timer : 90000
Running VN-Port Timer    : 213480
FCID                     : 0x2c1a03
WWN                      : 21:00:00:c0:dd:11:09:13
<table>
<thead>
<tr>
<th>VN-Port-MAC</th>
<th>: 0e:fc:00:03:00:03</th>
</tr>
</thead>
<tbody>
<tr>
<td>Session state</td>
<td>: Up</td>
</tr>
<tr>
<td>Configured FKA-ADV</td>
<td>: 90000</td>
</tr>
<tr>
<td>Running FKA-ADV</td>
<td>: 0</td>
</tr>
<tr>
<td>Configured VN-Port Timer</td>
<td>: 90000</td>
</tr>
<tr>
<td>Running VN-Port Timer</td>
<td>: 214004</td>
</tr>
<tr>
<td>FCID</td>
<td>: 0x2c1a04</td>
</tr>
<tr>
<td>WWN</td>
<td>: 21:00:00:c0:df:12:08:14</td>
</tr>
</tbody>
</table>
show fibre-channel fip enode

Syntax

show fibre-channel fip enode enode-mac
  <brief | detail>
  <vn-port-mac vn-port-mac>

Release Information
Command introduced in Junos OS Release 11.1 for the QFX Series.

Description
Display Fibre Channel over Ethernet (FCoE) Initialization Protocol (FIP) information for
a specified ENode or a specified VN_Port on an ENode.

Options
brief | detail—(Optional) Display the specified level of output.
enode-mac—Display information for the ENode specified by the MAC address.
vn-port-mac vn-port-mac—(Optional) Display information only for the specified VN_Port.

Required Privilege
view

Related Documentation
• Configuring FIP on an FCoE-FC Gateway
• show fibre-channel fip on page 277
• show fibre-channel fip fabric on page 286
• show fibre-channel fip fcf on page 289
• show fibre-channel fip interface on page 292
• show fibre-channel fip statistics on page 295
• clear fibre-channel fip enode on page 228

List of Sample Output
show fibre-channel fip enode on page 284
show fibre-channel fip enode detail on page 285

Output Fields
Table 25 on page 282 lists the output fields for the show fibre-channel fip enode command.
Output fields are listed in the approximate order in which they appear. A session is a
FLOGI or FDISC login to the FC SAN fabric. Session does not refer to end-to-end storage
sessions.

Table 25: show fibre-channel fip enode Output Fields

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
<th>Level of Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fabric Name</td>
<td>Name of the fabric and in parentheses the fabric ID.</td>
<td>All</td>
</tr>
<tr>
<td>FC-MAP</td>
<td>FCoE mapped address prefix of the FCoE forwarder for the fabric.</td>
<td>detail</td>
</tr>
<tr>
<td>FKA-ADV-PERIOD</td>
<td>Period of time in milliseconds between FIP keepalive advertisements configured for the FC fabric.</td>
<td>detail</td>
</tr>
</tbody>
</table>
Table 25: show fibre-channel fip enode Output Fields (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
<th>Level of Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAX-SESSIONS-PER-ENODE</td>
<td>Maximum number of concurrent sessions (FLOGI and FDISC combined) that each ENode can instantiate.</td>
<td>detail</td>
</tr>
<tr>
<td><strong>FCoE trusted</strong></td>
<td>Whether ports on the FC fabric are trusted or untrusted:</td>
<td>detail</td>
</tr>
<tr>
<td>• Yes</td>
<td>Ports on the FC fabric are trusted; FIP snooping is turned off.</td>
<td></td>
</tr>
<tr>
<td>• No</td>
<td>Ports on the FC fabric are not trusted; FIP snooping is turned on.</td>
<td></td>
</tr>
<tr>
<td><strong>Member</strong></td>
<td>Information about an FCF that is a member of the fabric.</td>
<td>All</td>
</tr>
<tr>
<td>• FCF-MAC</td>
<td>MAC address used in discovery advertisements.</td>
<td>All</td>
</tr>
<tr>
<td>• FKA-ADV-PERIOD</td>
<td>Period of time in milliseconds between FIP keepalive advertisements configured for the FC interface.</td>
<td>detail</td>
</tr>
<tr>
<td>• FKA-ADV-D-BIT</td>
<td>Disable FIP keepalive advertisement monitoring bit. The state is always off.</td>
<td>detail</td>
</tr>
<tr>
<td>• Type</td>
<td>Type of interface:</td>
<td>detail</td>
</tr>
<tr>
<td>• VF_Port Capable</td>
<td>Interface can act as a VF_Port interface.</td>
<td></td>
</tr>
<tr>
<td>• Priority</td>
<td>Priority value associated with the switch FCF-MAC. Converged network adapters (CNAs) use the priority value to determine the switch with which they will perform FIP FLOGI. The lower the value, the higher the priority. Value range: 0 through 255.</td>
<td>detail</td>
</tr>
<tr>
<td>• State</td>
<td>FIP state on the fabric:</td>
<td>detail</td>
</tr>
<tr>
<td>• Enable</td>
<td>FIP is enabled on the fabric.</td>
<td></td>
</tr>
<tr>
<td>• Disable</td>
<td>FIP is disabled on the fabric.</td>
<td></td>
</tr>
</tbody>
</table>
### Table 25: show fibre-channel fip enode Output Fields (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
<th>Level of Output</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ENode</strong></td>
<td>Information about a connected FCoE node (ENode).</td>
<td>All</td>
</tr>
<tr>
<td>• ENode-MAC</td>
<td>MAC address of the connected ENode.</td>
<td>All</td>
</tr>
<tr>
<td>• ENode State</td>
<td>Login state internal to Junos OS.</td>
<td>All</td>
</tr>
<tr>
<td>• Configured ENode timer</td>
<td>User-configured FIP keepalive advertisement interval in milliseconds.</td>
<td>detail</td>
</tr>
<tr>
<td>• Running ENode timer</td>
<td>Runtime interval in milliseconds of the last FIP keepalive advertisement received.</td>
<td>detail</td>
</tr>
<tr>
<td>• Active FIP Sessions</td>
<td>Number of active FIP sessions on the ENode.</td>
<td>detail</td>
</tr>
<tr>
<td>• VN-Port-MAC</td>
<td>MAC address of a VN_Port on the ENode.</td>
<td>detail</td>
</tr>
<tr>
<td>• Session State</td>
<td>Session state internal to Junos OS.</td>
<td>detail</td>
</tr>
<tr>
<td>• Configured FKA-ADV</td>
<td>User-configured FIP keepalive advertisement interval in milliseconds.</td>
<td>detail</td>
</tr>
<tr>
<td>• Running FKA-ADV</td>
<td>Runtime interval in seconds of the last FIP keepalive advertisement received.</td>
<td>detail</td>
</tr>
<tr>
<td>• Configured VN-Port Timer</td>
<td>Configured state of the VN_Port keepalive timer in seconds.</td>
<td>detail</td>
</tr>
<tr>
<td>• Running VN-Port Timer</td>
<td>Running state of the VN_Port keepalive timer in seconds.</td>
<td>detail</td>
</tr>
<tr>
<td>• FCID</td>
<td>Fibre Channel ID of the VN_Port.</td>
<td>detail</td>
</tr>
<tr>
<td>• WWN</td>
<td>Unique worldwide name of the VN_Port.</td>
<td>detail</td>
</tr>
</tbody>
</table>

### Sample Output

```
show fibre-channel fip enode

user@switch> show fibre-channel fip enode 00:10:94:00:00:02
Fabric Name : proxy2 (200)
Member
FC-F-MAC : 00:30:48:b0:ee:d2 (Interface vlan.100)
Enode
Enode-MAC : 00:10:94:00:00:02       State : Logged-in
Session
VN-Port-MAC : 0e:fc:00:03:00:02
VN-Port-MAC : 0e:fc:00:03:00:01
```
show fibre-channel fip enode detail

user@switch> show fibre-channel fip enode 00:10:94:00:00:02 detail
   Fabric Name : proxy2 (200)
   FC-MAP : 0e:fc:00
   FKA-ADV-PERIOD : 90000   MAX-SESSIONS-PER-ENODE : 32
   FCoE trusted : No

Member
   FCF-MAC: 00:30:48:b0:ee:d2  (Interface vlan.100)
   FKA-ADV-PERIOD : 90000    FKA-ADV-D-BIT-bit : Off
   Type : VF_Port Capable
   Priority : 86             State : Enable

ENode
   Enode-MAC : 00:10:94:00:00:02  ENode State : Logged-in
   Configured ENode timer: 8000    Running ENode timer: 12226
   Active FIP Sessions : 2

   Session details
   VN-Port-MAC : 0e:fc:00:03:00:02
   Session state : Up
   Configured FKA-ADV : 90000
   Running FKA-ADV : 0
   Configured VN-Port Timer : 90000
   Running VN-Port Timer : 213193
   FCID : 0x2c1a01
   WWN : 10:00:00:00:c9:a4:a3:cf

   VN-Port-MAC : 0e:fc:00:03:00:01
   Session state : Up
   Configured FKA-ADV : 90000
   Running FKA-ADV : 0
   Configured VN-Port Timer : 90000
   Running VN-Port Timer : 213632
   FCID : 0x2c1a02
   WWN : 10:00:00:00:d9:b4:e3:df
show fibre-channel fip fabric

Syntax  
show fibre-channel fip fabric fabric-name
       <brief | detail>

Release Information  
Command introduced in Junos OS Release 11.1 for the QFX Series.

Description  
Display Fibre Channel over Ethernet (FCoE) Initialization Protocol (FIP) information for a specified Fibre Channel fabric.

Options  
brief | detail—(Optional) Display the specified level of output.

fabric-name—Display information for the specified fabric.

Required Privilege  
view

Related Documentation  
- Configuring FIP on an FCoE-FC Gateway
- show fibre-channel fip on page 277
- show fibre-channel fip enode on page 282
- show fibre-channel fip fcf on page 289
- show fibre-channel fip interface on page 292
- show fibre-channel fip statistics on page 295

List of Sample Output  
show fibre-channel fip fabric proxy2 on page 287
show fibre-channel fip fabric detail on page 288

Output Fields  
Table 26 on page 286 lists the output fields for the show fibre-channel fip fabric command. Output fields are listed in the approximate order in which they appear.

Table 26: show fibre-channel fip fabric Output Fields

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
<th>Level of Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fabric Name</td>
<td>Name of the fabric and in parentheses the fabric ID.</td>
<td>All</td>
</tr>
<tr>
<td>FC-MAP</td>
<td>FCoE mapped address prefix of the FCoE forwarder for the fabric.</td>
<td>detail</td>
</tr>
<tr>
<td>FKA-ADV-PERIOD</td>
<td>Period of time in milliseconds between FIP keepalive advertisements configured for the FC fabric.</td>
<td>detail</td>
</tr>
</tbody>
</table>
Table 26: show fibre-channel fip fabric Output Fields (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
<th>Level of Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Member</td>
<td>Information about an FCF that is a member of the fabric.</td>
<td>All</td>
</tr>
<tr>
<td>FCF-MAC</td>
<td>MAC address used in discovery advertisements.</td>
<td>All</td>
</tr>
<tr>
<td>FKA-ADV-PERIOD</td>
<td>Period of time in milliseconds between FIP keepalive advertisements configured for the FC interface.</td>
<td>detail</td>
</tr>
<tr>
<td>FKA-ADV-D-BIT</td>
<td>Disable FIP keepalive advertisement monitoring bit. The state is always off.</td>
<td>detail</td>
</tr>
<tr>
<td>Type</td>
<td>Type of interface:</td>
<td>detail</td>
</tr>
<tr>
<td>VF_Port Capable</td>
<td>Interface can act as a VF_Port interface.</td>
<td></td>
</tr>
<tr>
<td>ENode</td>
<td>Information about a connected FCoE node (ENode).</td>
<td>All</td>
</tr>
<tr>
<td>ENode-MAC</td>
<td>MAC address of the connected ENode.</td>
<td>All</td>
</tr>
<tr>
<td>State</td>
<td>Login state internal to Junos OS.</td>
<td>All</td>
</tr>
<tr>
<td>VN-Port-MAC</td>
<td>MAC address of a VN_Port on the ENode.</td>
<td>detail</td>
</tr>
<tr>
<td>Session State</td>
<td>Session state internal to Junos OS.</td>
<td>detail</td>
</tr>
<tr>
<td>Configured FKA-ADV</td>
<td>User-configured FIP keepalive advertisement interval in milliseconds.</td>
<td>detail</td>
</tr>
<tr>
<td>Running FKA-ADV</td>
<td>Runtime interval in seconds of the last FIP keepalive advertisement received. This value changes every time an FKA_ADV is received.</td>
<td>detail</td>
</tr>
<tr>
<td>Configured VN-Port Timer</td>
<td>Configured state of the VN_Port keepalive timer in seconds. This value is always 90 and is not user-configurable.</td>
<td>detail</td>
</tr>
<tr>
<td>Running VN-Port Timer</td>
<td>Running state of the VN_Port keepalive timer in seconds.</td>
<td>detail</td>
</tr>
</tbody>
</table>

Sample Output

```
show fibre-channel fip fabric proxy2

user@switch> show fibre-channel fip fabric proxy2
Fabric Name : proxy2 (200)
Member
  FCF-MAC : 00:30:48:b0:ee:d2 (Interface vlan.100)
  ENode
```
Enode-MAC : 00:10:94:00:00:02       State : Logged-in
Enode-MAC : 00:10:94:00:00:03       State : Logged-in

show fibre-channel fip fabric detail

user@switch>  show fibre-channel fip fabric proxy2 detail
Fabric Name : proxy2 (200)
  FC-MAP         : 0e:fc:00
  FKA-ADV-PERIOD : 90000

Member
  FCF-MAC: 00:30:48:b0:ee:d2  (Interface vlan.100)
  FKA-ADV-PERIOD  : 90000     FKA-ADV-D-bit : Off
  Type : VF_Port Capable

ENode
  Enode-MAC : 00:10:94:00:00:02   State : Logged-in

    Session details
    VN-Port-MAC              : 0e:fc:00:03:00:02
    Session state            : Up
    Configured FKA-ADV       : 90000
    Running FKA-ADV          : 0
    Configured VN-Port Timer : 90
    Running VN-Port Timer    : 0

  VN-Port-MAC              : 0e:fc:00:03:00:01
  Session state            : Up
  Configured FKA-ADV       : 90000
  Running FKA-ADV          : 0
  Configured VN-Port Timer : 90
  Running VN-Port Timer    : 0

ENode
  Enode-MAC : 00:10:94:00:00:03   State : Logged-in

    Session details
    VN-Port-MAC              : 0e:fc:00:03:00:04
    Session state            : Up
    Configured FKA-ADV       : 90000
    Running FKA-ADV          : 0
    Configured VN-Port Timer : 90
    Running VN-Port Timer    : 0

  VN-Port-MAC              : 0e:fc:00:03:00:03
  Session state            : Up
  Configured FKA-ADV       : 90000
  Running FKA-ADV          : 0
  Configured VN-Port Timer : 90
  Running VN-Port Timer    : 0
show fibre-channel fip fcf

Syntax
show fibre-channel fip fcf fcf-mac
    <brief | detail>
    <fabric fabric-name>

Release Information
Command introduced in Junos OS Release 11.1 for the QFX Series.

Description
Display Fibre Channel over Ethernet (FCoE) Initialization Protocol (FIP) information for a specified FCoE forwarder (FCF).

Options
brief | detail—(Optional) Display the specified level of output.


fcf-mac—Display information for the FCF specified by the MAC address.

Required Privilege
view

Related Documentation
• Configuring FIP on an FCoE-FC Gateway
• show fibre-channel fip on page 277
• show fibre-channel fip enode on page 282
• show fibre-channel fip fabric on page 286
• show fibre-channel fip interface on page 292
• show fibre-channel fip statistics on page 295

List of Sample Output
show fibre-channel fip fcf on page 290
show fibre-channel fip fcf detail on page 291

Output Fields
Table 27 on page 289 lists the output fields for the show fibre-channel fip fcf command. Output fields are listed in the approximate order in which they appear.

Table 27: show fibre-channel fip fcf Output Fields

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
<th>Level of Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fabric Name</td>
<td>Name of the fabric and in parentheses the fabric ID.</td>
<td>All</td>
</tr>
<tr>
<td>FC-MAP</td>
<td>FCoE mapped address prefix of the FCoE forwarder for the fabric.</td>
<td>detail</td>
</tr>
<tr>
<td>FKA-ADV-PERIOD</td>
<td>Period of time in milliseconds between FIP keepalive advertisements configured for the FC fabric.</td>
<td>detail</td>
</tr>
</tbody>
</table>
Table 27: show fibre-channel fip fcf Output Fields (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
<th>Level of Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Member</td>
<td>Information about an FCF that is a member of the fabric.</td>
<td>All</td>
</tr>
<tr>
<td>• FCF-MAC</td>
<td>MAC address used in discovery advertisements.</td>
<td>All</td>
</tr>
<tr>
<td>• FKA-ADV-PERIOD</td>
<td>Period of time in milliseconds between FIP keepalive advertisements configured for the FC interface.</td>
<td>detail</td>
</tr>
<tr>
<td>• FKA-ADV-D-BIT</td>
<td>Disable FIP keepalive advertisement monitoring bit. The state is always off.</td>
<td>detail</td>
</tr>
<tr>
<td>• Type</td>
<td>Type of interface:</td>
<td>detail</td>
</tr>
<tr>
<td>• VF_Port Capable—Interface can act as a VF_Port interface.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ENode</td>
<td>Information about a connected FCoE node (ENode).</td>
<td>All</td>
</tr>
<tr>
<td>• ENode-MAC</td>
<td>MAC address of the connected ENode.</td>
<td>All</td>
</tr>
<tr>
<td>• State</td>
<td>Login state internal to Junos OS.</td>
<td>All</td>
</tr>
<tr>
<td>• VN-Port-MAC</td>
<td>MAC address of a VN_Port on the ENode.</td>
<td>detail</td>
</tr>
<tr>
<td>• Session State</td>
<td>Session state internal to Junos OS.</td>
<td>detail</td>
</tr>
<tr>
<td>• Configured FKA-ADV</td>
<td>User-configured FIP keepalive advertisement interval in milliseconds.</td>
<td>detail</td>
</tr>
<tr>
<td>• Running FKA-ADV</td>
<td>Runtime interval in seconds of the last FIP keepalive advertisement received. This value changes every time an FKA_ADV is received.</td>
<td>detail</td>
</tr>
<tr>
<td>• Configured VN-Port Timer</td>
<td>Configured state of the VN_Port keepalive timer in seconds. This value is always 90 and is not user-configurable.</td>
<td>detail</td>
</tr>
<tr>
<td>• Running VN-Port Timer</td>
<td>Running state of the VN_Port keepalive timer in seconds.</td>
<td>detail</td>
</tr>
</tbody>
</table>

Sample Output

show fibre-channel fip fcf

user@switch> show fibre-channel fip fcf 00:30:48:b0:ee:d2
Fabric Name : proxy2 (200)
Member
FCF-MAC : 00:30:48:b0:ee:d2 (Interface vlan.100)
ENode
show fibre-channel fip fcf detail

```
user@switch> show fibre-channel fip fcf 00:30:48:b0:ee:d2 detail
Fabric Name : proxy2 (200)
FC-MAP          : 0e:fc:00
FKA-ADV-PERIOD : 90000

Member
FCF-MAC: 00:30:48:b0:ee:d2  (Interface vlan.100)
FKA-ADV-PERIOD  : 90000     FKA-ADV-D-bit : Off
Type : VF_Port Capable

ENode
Enode-MAC : 00:10:94:00:00:02   State : Logged-in

Session details
VN-Port-MAC              : 0e:fc:00:03:00:02
Session state            : Up
Configured FKA-ADV       : 90000
Running FKA-ADV          : 0
Configured VN-Port Timer : 90
Running VN-Port Timer    : 0

VN-Port-MAC              : 0e:fc:00:03:00:01
Session state            : Up
Configured FKA-ADV       : 90000
Running FKA-ADV          : 0
Configured VN-Port Timer : 90
Running VN-Port Timer    : 0

ENode
Enode-MAC : 00:10:94:00:00:03   State : Logged-in

Session details
VN-Port-MAC              : 0e:fc:00:03:00:04
Session state            : Up
Configured FKA-ADV       : 90000
Running FKA-ADV          : 0
Configured VN-Port Timer : 90
Running VN-Port Timer    : 0

VN-Port-MAC              : 0e:fc:00:03:00:03
Session state            : Up
Configured FKA-ADV       : 90000
Running FKA-ADV          : 0
Configured VN-Port Timer : 90
Running VN-Port Timer    : 0
```
show fibre-channel fip interface

Syntax

```
show fibre-channel fip interface interface-name
  <brief | detail>
  <enode enode-mac>
  <fabric fabric-name>
  <vn-port vn-port-mac>
```

Release Information

Command introduced in Junos OS Release 11.1 for the QFX Series.

Description

Display Fibre Channel over Ethernet (FCoE) Initialization Protocol (FIP) information for a specified interface.

Options

- `brief | detail`—(Optional) Display the specified level of output.
- `enode-mac`—MAC address of the ENode.
- `interface-name`—Display information for the specified interface.
- `vn-port-mac`—MAC address of the VN_Port.

Required Privilege Level

`view`

Related Documentation

- Configuring FIP on an FCoE-FC Gateway
- show fibre-channel fip on page 277
- show fibre-channel fip enode on page 282
- show fibre-channel fip fabric on page 286
- show fibre-channel fip fcf on page 289
- show fibre-channel fip statistics on page 295
- clear fibre-channel fip vn-port on page 230

List of Sample Output

- show fibre-channel fip interface on page 293
- show fibre-channel fip interface detail on page 294

Output Fields

Table 28 on page 292 lists the output fields for the `show fibre-channel fip interface` command. Output fields are listed in the approximate order in which they appear.

Table 28: show fibre-channel fip interface Output Fields

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
<th>Level of Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fabric Name</td>
<td>Name of the fabric and in parentheses the fabric ID.</td>
<td>All</td>
</tr>
<tr>
<td>FC-MAP</td>
<td>FCoE mapped address prefix of the FCoE forwarder for the fabric.</td>
<td>detail</td>
</tr>
</tbody>
</table>
### Table 28: show fibre-channel fip interface Output Fields (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
<th>Level of Output</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FKA-ADV-PERIOD</strong></td>
<td>Period of time in milliseconds between FIP keepalive advertisements configured for the FC fabric.</td>
<td>detail</td>
</tr>
<tr>
<td><strong>Member</strong></td>
<td>Information about an FCF that is a member of the fabric.</td>
<td>All</td>
</tr>
<tr>
<td></td>
<td>• FCF-MAC</td>
<td>All</td>
</tr>
<tr>
<td></td>
<td>MAC address used in discovery advertisements.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• FKA-ADV-PERIOD</td>
<td>detail</td>
</tr>
<tr>
<td></td>
<td>Period of time in milliseconds between FIP keepalive advertisements configured for the FC interface.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• FKA-ADV-D-BIT</td>
<td>detail</td>
</tr>
<tr>
<td></td>
<td>Disable FIP keepalive advertisement monitoring bit. The state is always off.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Type</td>
<td>detail</td>
</tr>
<tr>
<td></td>
<td>Type of interface:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• VF_Port Capable—Interface can act as a VF_Port interface.</td>
<td></td>
</tr>
<tr>
<td><strong>ENode</strong></td>
<td>Information about a connected FCoE node (ENode).</td>
<td>All</td>
</tr>
<tr>
<td></td>
<td>• ENode-MAC</td>
<td>All</td>
</tr>
<tr>
<td></td>
<td>MAC address of the connected ENode.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• State</td>
<td>All</td>
</tr>
<tr>
<td></td>
<td>Login state internal to Junos OS.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• VN-Port-MAC</td>
<td>detail</td>
</tr>
<tr>
<td></td>
<td>MAC address of a VN_Port on the ENode.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Session State</td>
<td>detail</td>
</tr>
<tr>
<td></td>
<td>Session state internal to Junos OS.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Configured FKA-ADV</td>
<td>detail</td>
</tr>
<tr>
<td></td>
<td>User-configured FIP keepalive advertisement interval in milliseconds.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Running FKA-ADV</td>
<td>detail</td>
</tr>
<tr>
<td></td>
<td>Runtime interval in seconds of the last FIP keepalive advertisement received. This value changes every time an FKAADV is received.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Configured VN-Port Timer</td>
<td>detail</td>
</tr>
<tr>
<td></td>
<td>Configured state of the VN_Port keepalive timer in seconds. This value is always 90 and is not user-configurable.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Running VN-Port Timer</td>
<td>detail</td>
</tr>
<tr>
<td></td>
<td>Running state of the VN_Port keepalive timer in seconds.</td>
<td></td>
</tr>
</tbody>
</table>

### Sample Output
show fibre-channel fip interface

```
user@switch> show fibre-channel fip interface vlan.100
Fabric Name : proxy2 (200)
Member
```
show fibre-channel fip interface detail

user@switch> show fibre-channel fip interface vlan.100 detail
Fabric Name : proxy2 (200)
FC-MAP         : 0e:fc:00
FKA-ADV-PERIOD : 90000

Member
FCF-MAC : 00:30:48:b0:ee:d2 (Interface vlan.100)
FKA-ADV-PERIOD  : 90000     FKA-ADV-D-bit : Off
Type : VF_Port Capable

ENode
Enode-MAC : 00:10:94:00:00:02   State : Logged-in

Session details
VN-Port-MAC              : 0e:fc:00:03:00:02
Session state            : Up
Configured FKA-ADV       : 90000
Running FKA-ADV          : 0
Configured VN-Port Timer : 90
Running VN-Port Timer    : 0

ENode
Enode-MAC : 00:10:94:00:00:03   State : Logged-in

Session details
VN-Port-MAC              : 0e:fc:00:03:00:01
Session state            : Up
Configured FKA-ADV       : 90000
Running FKA-ADV          : 0
Configured VN-Port Timer : 90
Running VN-Port Timer    : 0

Copyright © 2014, Juniper Networks, Inc.
**show fibre-channel fip statistics**

**Syntax**  
`show fibre-channel fip statistics`  
`<fabric fabric-name>`

**Release Information**  
Command introduced in Junos OS Release 11.1 for the QFX Series.

**Description**  
Display Fibre Channel over Ethernet Initialization Protocol (FIP) statistics.

**Options**  

**Required Privilege Level**  
view

**Related Documentation**  
- show fibre-channel fip on page 277  
- show fibre-channel fip enode on page 282  
- show fibre-channel fip fabric on page 286  
- show fibre-channel fip fcf on page 289  
- show fibre-channel fip interface on page 292  
- clear fibre-channel fip statistics on page 229

**List of Sample Output**  
*show fibre-channel fip statistics on page 297*

**Output Fields**  
Table 29 on page 295 lists the output fields for the show fibre-channel fip statistics command. Output fields are listed in the approximate order in which they appear.

**Table 29: show fibre-channel fip statistics Output Fields**

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fabric name</td>
<td>Name of the fabric.</td>
</tr>
<tr>
<td>Interface name</td>
<td>Name of the FCoE VLAN interface.</td>
</tr>
</tbody>
</table>

Copyright © 2014, Juniper Networks, Inc.
### Table 29: show fibre-channel fip statistics Output Fields (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FIP Message Type</strong></td>
<td>Type of FIP message for the displayed row of statistics.</td>
</tr>
<tr>
<td>MDS</td>
<td>Number of multicast discovery solicitations.</td>
</tr>
<tr>
<td>UDS</td>
<td>Number of unicast discovery solicitations.</td>
</tr>
<tr>
<td>FLOGI</td>
<td>Number of fabric login (FLOGI) messages.</td>
</tr>
<tr>
<td>FDISC</td>
<td>Number of fabric discovery (FDISC) messages.</td>
</tr>
<tr>
<td>LOGO</td>
<td>Number of fabric logout (LOGO) messages.</td>
</tr>
<tr>
<td>ENODE KA</td>
<td>Number of ENode keepalive messages.</td>
</tr>
<tr>
<td>VN_Port KA</td>
<td>Number of VN_Port keepalive messages.</td>
</tr>
<tr>
<td>MDA</td>
<td>Number of multicast discovery advertisements.</td>
</tr>
<tr>
<td>UDA</td>
<td>Number of unicast discovery advertisements.</td>
</tr>
<tr>
<td>FLOGI ACC</td>
<td>Number of fabric login requests accepted.</td>
</tr>
<tr>
<td>FLOGI RJT</td>
<td>Number of fabric login requests rejected.</td>
</tr>
<tr>
<td>FDISC ACC</td>
<td>Number of fabric discovery requests accepted.</td>
</tr>
<tr>
<td>FDISC RJT</td>
<td>Number of fabric discovery requests rejected.</td>
</tr>
<tr>
<td>LOGO ACC</td>
<td>Number of logout requests accepted.</td>
</tr>
<tr>
<td>LOGO RJT</td>
<td>Number of logout requests rejected.</td>
</tr>
<tr>
<td>CVL</td>
<td>Number of clear virtual links (CVL) messages.</td>
</tr>
<tr>
<td>CVL ALL</td>
<td>Number of CVL all messages.</td>
</tr>
<tr>
<td>Received</td>
<td>Number of messages received.</td>
</tr>
<tr>
<td>Sent</td>
<td>Number of messages sent.</td>
</tr>
<tr>
<td>Rx errors</td>
<td>Number of receive errors.</td>
</tr>
</tbody>
</table>
Table 29: show fibre-channel fip statistics Output Fields (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dropped</td>
<td>Number of dropped messages.</td>
</tr>
<tr>
<td></td>
<td><strong>NOTE:</strong> One cause of dropped messages is that the system limits the number of discovery solicitations (MDS and UDS) it accepts to a maximum of 100 outstanding requests at any given time. If the system has 100 discovery solicitations outstanding, the system does not respond to new discovery solicitations. Instead, the system drops new discovery solicitations and reports the number of dropped discovery solicitations in this field. When there are fewer than 100 outstanding discovery solicitations, the system responds to new requests as usual with a discovery advertisement.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>General Statistics</th>
<th>Number of frames recvd with invalid src-mac</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Number of frames received that have an invalid source media access control (MAC) address.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Number of frames recvd with invalid version</th>
<th>Number of FIP frames received with an Invalid FIP version.</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Number of frames recvd with invalid opcode</th>
<th>Number of FIP validation descriptors with an invalid opcode received.</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Number of frames recvd with invalid subcode</th>
<th>Number of FIP validation descriptors with an invalid subcode received.</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Number of frames recvd on inactive FCF</th>
<th>Number of frames received on a logical interface if FIP is not active on that logical interface (for example, if a WWN is not allocated to that logical interface).</th>
</tr>
</thead>
</table>

Sample Output

show fibre-channel fip statistics

```
user@switch> show fibre-channel fip statistics
Fabric name: proxy2

Interface name: vlan.100

<table>
<thead>
<tr>
<th>FIP Message type</th>
<th>Received</th>
<th>Sent</th>
<th>Rx errors</th>
<th>Dropped</th>
</tr>
</thead>
<tbody>
<tr>
<td>MDS</td>
<td>22236</td>
<td>0</td>
<td>0</td>
<td>17089</td>
</tr>
<tr>
<td>UDS</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>FLOGI</td>
<td>1257</td>
<td>0</td>
<td>8</td>
<td>0</td>
</tr>
<tr>
<td>FDISC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>LOGO</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>ENODE KA</td>
<td>455</td>
<td>0</td>
<td>6</td>
<td>0</td>
</tr>
<tr>
<td>VN_Port KA</td>
<td>22</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>MDA</td>
<td>0</td>
<td>243</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>UDA</td>
<td>0</td>
<td>5147</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>FLOGI ACC</td>
<td>0</td>
<td>376</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>FLOGI RJT</td>
<td>0</td>
<td>881</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>FDISC ACC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>FDISC RJT</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>LOGO</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>LOGO ACC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>LOGO RJT</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
```
CVL                   0            374      0           0
CVL ALL               0            380      0           0

General Statistics:

Number of frame recvd with invalid src-mac: 0
Number of frame recvd with invalid version: 0
Number of frame recvd with invalid opcode: 0
Number of frame recvd with invalid subcode: 0
Number of frame recvd on inactive FCF: 0
show fibre-channel flogi fport

Syntax
show fibre-channel flogi fport
<fabric fabric-name>

Release Information
Command introduced in Junos OS Release 11.1 for the QFX Series.

Description
Display Fibre Channel fabric login (FLOGI) F_Port information.

Options

Required Privilege Level
view

Related Documentation
• show fibre-channel flogi nport on page 301
• show fibre-channel flogi statistics on page 303

List of Sample Output
show fibre-channel flogi fport on page 299

Output Fields
Table 30 on page 299 lists the output fields for the show fibre-channel flogi fport command. Output fields are listed in the approximate order in which they appear.

Table 30: show fibre-channel flogi fport Output Fields

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fabric</td>
<td>Name of the fabric.</td>
</tr>
<tr>
<td>Interface</td>
<td>Name of the switch VF_Port interface.</td>
</tr>
<tr>
<td>Mac-Address</td>
<td>Media access control (MAC) address of the ENode.</td>
</tr>
<tr>
<td>State</td>
<td>Interface physical state: up or down.</td>
</tr>
<tr>
<td>Logins</td>
<td>Number of logins to the VF_Port.</td>
</tr>
<tr>
<td>NPIV</td>
<td>N_Port ID virtualization (NPIV) state: Yes or No.</td>
</tr>
<tr>
<td>FLOGI-Port-WWN</td>
<td>Unique worldwide name (WWN) of the VN_Port performing fabric login (FLOGI) to the switch VF_Port.</td>
</tr>
</tbody>
</table>

Sample Output
show fibre-channel flogi fport

user@switch> show fibre-channel flogi fport
Fabric: proxy2

<table>
<thead>
<tr>
<th>Interface</th>
<th>Mac-Address</th>
<th>State</th>
<th>Logins</th>
<th>NPIV</th>
<th>FLOGI-Port-WWN</th>
</tr>
</thead>
<tbody>
<tr>
<td>vlan.100</td>
<td>00:10:94:00:00:02</td>
<td>Up</td>
<td>2</td>
<td>Yes</td>
<td>20:00:10:94:00:01:00:01</td>
</tr>
<tr>
<td>vlan.100</td>
<td>00:10:94:00:00:03</td>
<td>Up</td>
<td>2</td>
<td>Yes</td>
<td>20:00:10:94:00:02:00:01</td>
</tr>
</tbody>
</table>
show fibre-channel flogi nport

Syntax

show fibre-channel flogi nport
    <brief | detail>
    <fabric fabric-name>

Release Information

Command introduced in Junos OS Release 11.1 for the QFX Series.

Description

Display Fibre Channel fabric login (FLOGI) VN_Port information.

Options

brief | detail—(Optional) Display the specified level of output.


Required Privilege

view

Related Documentation

- show fibre-channel flogi fport on page 299
- show fibre-channel flogi statistics on page 303

List of Sample Output

show fibre-channel flogi nport on page 302
show fibre-channel flogi nport detail on page 302

Output Fields

Table 31 on page 301 lists the output fields for the show fibre-channel flogi nport command. Output fields are listed in the approximate order in which they appear.

Table 31: show fibre-channel flogi nport Output Fields

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
<th>Level of Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fabric</td>
<td>Name of the fabric.</td>
<td>All</td>
</tr>
<tr>
<td>Virtual-switch</td>
<td>Name of the fabric.</td>
<td>detail</td>
</tr>
<tr>
<td>Interface</td>
<td>Name of the VF_Port interface.</td>
<td>All</td>
</tr>
<tr>
<td>FCID</td>
<td>VN_Port Fibre Channel identifier provided by the Fibre Channel over Ethernet Forwarder (FCoE forwarder) or the Fibre Channel switch.</td>
<td>All</td>
</tr>
<tr>
<td>Port-WWN</td>
<td>Unique worldwide name (WWN) of the VN_Port.</td>
<td>All</td>
</tr>
<tr>
<td>Node-WWN</td>
<td>Unique WWN of the node hosting the VN_Port.</td>
<td>All</td>
</tr>
<tr>
<td>State or Flogi-state</td>
<td>Login state internal to Junos OS.</td>
<td>All</td>
</tr>
<tr>
<td>FLOGI-Port-WWN</td>
<td>Unique worldwide name (WWN) of the VN_Port performing fabric login (FLOGI) to the switch VF_Port.</td>
<td>detail</td>
</tr>
</tbody>
</table>
Sample Output

show fibre-channel flogi nport

user@switch> show fibre-channel flogi nport
Fabric: proxy2
<table>
<thead>
<tr>
<th>Interface</th>
<th>FCID</th>
<th>Port-WWN</th>
<th>Node-WWN</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>vlan.100</td>
<td>0x030001</td>
<td>20:00:10:94:00:01:00:01</td>
<td>10:00:10:94:00:00:00:01</td>
<td>online</td>
</tr>
<tr>
<td>vlan.100</td>
<td>0x030002</td>
<td>20:00:10:94:00:01:00:05</td>
<td>10:00:10:94:00:00:00:01</td>
<td>online</td>
</tr>
<tr>
<td>vlan.100</td>
<td>0x030003</td>
<td>20:00:10:94:00:02:00:01</td>
<td>10:00:10:94:00:00:00:01</td>
<td>online</td>
</tr>
<tr>
<td>vlan.100</td>
<td>0x030004</td>
<td>20:00:10:94:00:02:00:05</td>
<td>10:00:10:94:00:00:00:02</td>
<td>online</td>
</tr>
</tbody>
</table>

show fibre-channel flogi nport detail

user@switch> show fibre-channel flogi nport detail
Fabric: proxy2
Virtual-switch: proxy2

   Interface: vlan.100
   Flogi-state: online
   FCID: 0x030001
   Port-WWN: 20:00:10:94:00:01:00:01
   Node-WWN: 10:00:10:94:00:00:00:01
   FLOGI-Port-WWN: 20:00:10:94:00:01:00:01

   Interface: vlan.100
   Flogi-state: online
   FCID: 0x030002
   Port-WWN: 20:00:10:94:00:01:00:05
   Node-WWN: 10:00:10:94:00:00:00:01
   FLOGI-Port-WWN: 20:00:10:94:00:01:00:01

   Interface: vlan.100
   Flogi-state: online
   FCID: 0x030003
   Port-WWN: 20:00:10:94:00:02:00:01
   Node-WWN: 10:00:10:94:00:00:00:02
   FLOGI-Port-WWN: 20:00:10:94:00:02:00:01

   Interface: vlan.100
   Flogi-state: online
   FCID: 0x030004
   Port-WWN: 20:00:10:94:00:02:00:05
   Node-WWN: 10:00:10:94:00:00:00:02
   FLOGI-Port-WWN: 20:00:10:94:00:02:00:01
show fibre-channel flogi statistics

Syntax
show fibre-channel flogi statistics
    <fabric fabric-name>

Release Information
Command introduced in Junos OS Release 11.1 for the QFX Series.

Description
Display Fibre Channel fabric login (FLOGI) statistics.

Options

Required Privilege Level
view

Related Documentation
• show fibre-channel flogi fport on page 299
• show fibre-channel flogi nport on page 301
• clear fibre-channel flogi statistics on page 231

List of Sample Output
show fibre-channel flogi statistics on page 304

Output Fields
Table 32 on page 303 lists the output fields for the show fibre-channel flogi statistics command. Output fields are listed in the approximate order in which they appear.

Table 32: show fibre-channel flogi statistics Output Fields

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fabric</td>
<td>Name of the fabric.</td>
</tr>
<tr>
<td>FLOGI-Server Message type</td>
<td>Type of message:</td>
</tr>
<tr>
<td></td>
<td>• FLOGI—Fabric login (FLOGI) messages.</td>
</tr>
<tr>
<td></td>
<td>• FDISC—Fabric discovery (FDISC) messages.</td>
</tr>
<tr>
<td></td>
<td>• FLOGO—Fabric logout messages.</td>
</tr>
<tr>
<td></td>
<td>• FLOGO-LS-ACC—Fabric logout link service accept messages.</td>
</tr>
<tr>
<td></td>
<td>• LS-Accept—Link service accept messages.</td>
</tr>
<tr>
<td></td>
<td>• LS-Reject—Link service reject messages.</td>
</tr>
<tr>
<td></td>
<td>• invalid—Invalid messages.</td>
</tr>
<tr>
<td>Received</td>
<td>Number of messages received for a given message type.</td>
</tr>
<tr>
<td>Sent</td>
<td>Number of messages sent for a given message type.</td>
</tr>
<tr>
<td>Fabric</td>
<td>Name of the fabric.</td>
</tr>
<tr>
<td>Rx errors</td>
<td>Number of receive errors for a given type of message.</td>
</tr>
</tbody>
</table>
Table 32: show fibre-channel flogi statistics Output Fields (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>General Statistics</strong></td>
<td></td>
</tr>
<tr>
<td>• Number of FC2 Header Parse Errors</td>
<td>Number of errors parsing the FC-2 header.</td>
</tr>
<tr>
<td>• Number of FLOGI Parse Errors</td>
<td>Number of errors parsing fabric login requests.</td>
</tr>
<tr>
<td>• Number of FDISC Parse Errors</td>
<td>Number of errors parsing fabric discovery requests.</td>
</tr>
<tr>
<td>• Number of FLOGO Parse Errors</td>
<td>Number of errors parsing fabric logout requests.</td>
</tr>
<tr>
<td>• Number of Logins Discarded as Domain-ID not available</td>
<td>Number of discarded logins due to unavailability of a domain ID.</td>
</tr>
<tr>
<td>• Number of Logins Discarded as FCID not available</td>
<td>Number of discarded logins due to the unavailability of a Fibre Channel ID.</td>
</tr>
<tr>
<td>• Number of FCID requests deferred</td>
<td>Number of deferred FCID requests.</td>
</tr>
<tr>
<td>• Number of deferred FCID requests failed</td>
<td>Number of deferred FCID requests that failed.</td>
</tr>
</tbody>
</table>

Sample Output

show fibre-channel flogi statistics

user@switch> show fibre-channel flogi statistics
  Fabric: proxy2

<table>
<thead>
<tr>
<th>FLOGI-Server Message type</th>
<th>Received</th>
<th>Sent</th>
<th>Rx errors</th>
</tr>
</thead>
<tbody>
<tr>
<td>FLOGI</td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>FDISC</td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>FLOGO</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>FLOGO-LS-ACC</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>LS-Accept</td>
<td>0</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>LS-Reject</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>invalid</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

General Statistics:

Number of FC2 Header Parse Errors: 0
Number of FLOGI Parse Errors: 0
Number of FDISC Parse Errors: 0
Number of FLOGO Parse Errors: 0
Number of Logins Discarded as Domain-ID not available: 0
Number of Logins Discarded as FCID not available: 0
Number of FCID requests deferred: 0
Number of deferred FCID requests failed: 0
show fibre-channel interfaces

Syntax

```
brief | detail
fabric fabric-name
show fibre-channel interfaces interface-name
```

Release Information

Command introduced in Junos OS Release 11.1 for the QFX Series.

Description

Display information about Fibre Channel (FC) interfaces.

Options

- **brief | detail**—(Optional) Display the specified level of output.
- **interface-name**—Display output for the specified interface.

Required Privilege

view

Related Documentation

- Example: Setting Up Fibre Channel and FCoE VLAN Interfaces in an FCoE-FC Gateway Fabric
  - Configuring a Physical Fibre Channel Interface on page 170
  - Configuring an FCoE VLAN Interface on an FCoE-FC Gateway
  - Configuring a Fibre Channel Interface
  - Assigning Interfaces to a Fibre Channel Fabric

List of Sample Output

- show fibre-channel interfaces on page 307
- show fibre-channel interfaces detail on page 308

Output Fields

Table 33 on page 306 lists the output fields for the `show fibre-channel interfaces` command. Output fields are listed in the approximate order in which they appear.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
<th>Level of Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interface</td>
<td>Name of the FC interface.</td>
<td>All</td>
</tr>
<tr>
<td>Idx or Index</td>
<td>Interface index internal to Junos OS.</td>
<td>All</td>
</tr>
<tr>
<td>Type</td>
<td>Type of interface:</td>
<td>All</td>
</tr>
<tr>
<td></td>
<td>• FC—Native FC interface</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• FCOE—Fibre Channel over Ethernet interface</td>
<td></td>
</tr>
<tr>
<td>Native Fabric-id</td>
<td>Identification number of the QFX Series fabric.</td>
<td>All</td>
</tr>
<tr>
<td>NPIV</td>
<td>N_Port ID virtualization (NPIV) state: Yes or No.</td>
<td>All</td>
</tr>
</tbody>
</table>
Table 33: show fibre-channel interfaces Output Fields (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
<th>Level of Output</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Config-Mode</strong></td>
<td>User-configured port mode:</td>
<td>All</td>
</tr>
<tr>
<td></td>
<td>• <strong>F</strong>—The port is configured as a VF_Port, an FCoE port connected to FCoE devices.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• <strong>NP</strong>—The port is configured as a proxy N_Port (NP_Port), a native FC port connected to an FC switch.</td>
<td></td>
</tr>
<tr>
<td><strong>Oper-Mode</strong></td>
<td>Operational port mode:</td>
<td>All</td>
</tr>
<tr>
<td></td>
<td>• <strong>F</strong>—The port is operating as a VF_Port, an FCoE port connected to FCoE devices.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• <strong>NP</strong>—The port is operating as an NP_Port, a native FC port connected to an FC switch or an FCoE forwarder (FCF).</td>
<td></td>
</tr>
<tr>
<td><strong>State</strong></td>
<td>Interface state: up or down.</td>
<td>All</td>
</tr>
<tr>
<td><strong>WWN</strong></td>
<td>Unique worldwide name (WWN) of the port.</td>
<td>detail</td>
</tr>
<tr>
<td><strong>FSM-State</strong></td>
<td>Finite state machine state, internal to Junos OS.</td>
<td>detail</td>
</tr>
<tr>
<td><strong>Class ID</strong></td>
<td>Fibre Channel interface class ID, internal to Junos OS.</td>
<td>detail</td>
</tr>
<tr>
<td><strong>BB.SC.N</strong></td>
<td>Buffer-to-buffer state change number.</td>
<td>detail</td>
</tr>
<tr>
<td><strong>Tx B2B credits</strong></td>
<td>Number of buffer-to-buffer credits advertised by the neighbor switch that is connected to the FC interface.</td>
<td>detail</td>
</tr>
<tr>
<td><strong>Fabric</strong></td>
<td>Name of the fabric.</td>
<td>detail</td>
</tr>
<tr>
<td><strong>Remote-MAC</strong></td>
<td>Media access control (MAC) address of the remotely connected FCoE device VN_Port interface.</td>
<td>detail</td>
</tr>
<tr>
<td><strong>Tagging</strong></td>
<td>Not used. Value is shown as untagged.</td>
<td>detail</td>
</tr>
<tr>
<td><strong>Mode</strong></td>
<td>Logical interface (LIF) mode of operation.</td>
<td>detail</td>
</tr>
<tr>
<td><strong>H/W token</strong></td>
<td>Unique identifier for the FCoE VLAN interface, internal to Junos OS.</td>
<td>detail</td>
</tr>
</tbody>
</table>

**Sample Output**

```
user@switch> show fibre-channel interfaces

<table>
<thead>
<tr>
<th>Interface</th>
<th>Idx</th>
<th>Type</th>
<th>Native Fabric-id</th>
<th>NPIV</th>
<th>Config Mode</th>
<th>Oper Mode</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>fc-0/0/1.0</td>
<td>70</td>
<td>FC</td>
<td>200</td>
<td>YES</td>
<td>NP</td>
<td>NP</td>
<td>up</td>
</tr>
<tr>
<td>vlan.100</td>
<td>84</td>
<td>FCOE</td>
<td>200</td>
<td>YES</td>
<td>F</td>
<td>F</td>
<td>up</td>
</tr>
</tbody>
</table>
```
show fibre-channel interfaces detail

```
user@switch> show fibre-channel interfaces detail
Interface: fc-0/0/1.0, Index: 70, Type: FC, Native Fabric-id: 200
NPIV: YES, Config-Mode: NP, Oper-Mode: NP, State: up
WWN: 10:00:00:15:17:a9:98:64, FSM-State: up, Class ID: 1, BB_SC_N: 0
Tx B2B credits: 32

Fabric  Remote-MAC  Tagging  Mode  Oper state
proxy2  -           untagged  NP    up

Interface: vlan.100, Index: 84, Type: FCOE, Native Fabric-id: 200
NPIV: YES, Config-Mode: F, Oper-Mode: F, State: up
WWN: 10:00:00:30:48:b0:ee:d2, FSM-State: up
H/W token: 13

Fabric  Remote-MAC  Tagging  Mode  Oper state
proxy2  00:10:94:00:00:02  untagged  VF    up
proxy2  00:10:94:00:00:03  untagged  VF    up
```
show fibre-channel next-hops

Syntax
show fibre-channel next-hops

Release Information
Command introduced in Junos OS Release 11.1 for the QFX Series.

Description
Display Fibre Channel next-hop route information.

Required Privilege
view

Related Documentation
• show fibre-channel routes on page 311
• show route forwarding-table family fibre-channel on page 350

List of Sample Output
show fibre-channel next-hops on page 309

Output Fields
Table 34 on page 309 lists the output fields for the show fibre-channel next-hops command. Output fields are listed in the approximate order in which they appear.

Table 34: show fibre-channel next-hops Output Fields

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>Type of next hop internal to Junos OS.</td>
</tr>
<tr>
<td>State</td>
<td>State of the NP_Port interface:</td>
</tr>
<tr>
<td></td>
<td>• Active—The interface is online.</td>
</tr>
<tr>
<td></td>
<td>• Deleted—The interface is deleted.</td>
</tr>
<tr>
<td>Interface</td>
<td>Name of the interface.</td>
</tr>
<tr>
<td>Mac-Address</td>
<td>Media access control (MAC) address of the interface.</td>
</tr>
<tr>
<td>Index</td>
<td>Next-hop index identifier.</td>
</tr>
<tr>
<td>Ref-count</td>
<td>Reference count internal to Junos OS.</td>
</tr>
<tr>
<td>Flags</td>
<td>Flags internal to Junos OS.</td>
</tr>
</tbody>
</table>

Sample Output

show fibre-channel next-hops

```
user@switch> show fibre-channel next-hops
Type    State  Interface    Mac-Address         Index  Ref-count Flags
intf    Active fc-0/0/0.0                       0       1
ucast   Active vlan.100     00:15:17:a9:98:64   674     1        kernel, self
ucast   Active vlan.100     0e:fc:00:03:00:01   675     1        kernel, self
ucast   Active vlan.100     0e:fc:00:03:00:02   676     1        kernel, self
ucast   Active vlan.100     0e:fc:00:03:00:03   677     1        kernel, self
ucast   Active vlan.100     0e:fc:00:03:00:04   678     1        kernel, self
```
**show fibre-channel routes**

**Syntax**
```
show fibre-channel routes
<fabric fabric-name>
```

**Release Information**
Command introduced in Junos OS Release 11.1 for the QFX Series.

**Description**
Display Fibre Channel route information.

**Options**

**Required Privilege**
View

**Related Documentation**
- show fibre-channel next-hops on page 309
- show route forwarding-table family fibre-channel on page 350

**List of Sample Output**
show fibre-channel routes on page 311

**Output Fields**
Table 35 on page 311 lists the output fields for the `show fibre-channel routes` command. Output fields are listed in the approximate order in which they appear.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fabric</td>
<td>Name of the fabric.</td>
</tr>
<tr>
<td>Route-prefix</td>
<td>Route destination.</td>
</tr>
<tr>
<td>State</td>
<td>State of the NP_Port interface:</td>
</tr>
<tr>
<td></td>
<td>• Active—The interface is online.</td>
</tr>
<tr>
<td></td>
<td>• Deleted—The interface is deleted.</td>
</tr>
<tr>
<td>Interface</td>
<td>Name of the interface.</td>
</tr>
<tr>
<td>Mac-Address</td>
<td>Media access control (MAC) address of the interface.</td>
</tr>
<tr>
<td>Index</td>
<td>Next-hop index identifier.</td>
</tr>
<tr>
<td>Flags</td>
<td>Flags internal to Junos OS.</td>
</tr>
</tbody>
</table>

**Sample Output**

```
user@switch> show fibre-channel routes
Fabric: proxy2
Route-prefix 0x030000/24 State Active Interface fc-0/0/0.0 Mac-Address 00:15:17:a9:98:64 Index 674 Flags kernel
```
<table>
<thead>
<tr>
<th>Address</th>
<th>Status</th>
<th>VLAN</th>
<th>MAC Address</th>
<th>Index</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x030001/24</td>
<td>Active</td>
<td>vlan.100</td>
<td>0e:fc:00:03:00:01</td>
<td>675</td>
<td>kernel</td>
</tr>
<tr>
<td>0x030002/24</td>
<td>Active</td>
<td>vlan.100</td>
<td>0e:fc:00:03:00:02</td>
<td>676</td>
<td>kernel</td>
</tr>
<tr>
<td>0x030003/24</td>
<td>Active</td>
<td>vlan.100</td>
<td>0e:fc:00:03:00:03</td>
<td>677</td>
<td>kernel</td>
</tr>
<tr>
<td>0x030004/24</td>
<td>Active</td>
<td>vlan.100</td>
<td>0e:fc:00:03:00:04</td>
<td>678</td>
<td>kernel</td>
</tr>
</tbody>
</table>
show fibre-channel proxy fabric-state

Syntax
show fibre-channel proxy fabric-state
<fabric fabric-name>

Release Information
Command introduced in Junos OS Release 12.1 for the QFX Series.

Description
Display Fibre Channel (FC) proxy fabric state information.

Options

Required Privilege
view

Related Documentation
- Monitoring Fibre Channel Interface Load Balancing
- show fibre-channel proxy login-table on page 317
- show fibre-channel proxy np-port on page 320
- show fibre-channel proxy statistics on page 323

List of Sample Output
- show fibre-channel proxy fabric-state on page 315
- show fibre-channel proxy fabric-state fabric on page 315

Output Fields
Table 36 on page 313 lists the output fields for the show fibre-channel proxy fabric-state command. Output fields are listed in the approximate order in which they appear.

Table 36: show fibre-channel proxy fabric-state Output Fields

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fabric</td>
<td>Name of the fabric.</td>
</tr>
<tr>
<td>Fabric-id</td>
<td>Fabric ID number.</td>
</tr>
<tr>
<td>Proxy load balance algorithm</td>
<td>Load-balancing algorithm used on the FCoE-FC gateway FC fabric:</td>
</tr>
<tr>
<td></td>
<td>• Simple—Load balancing is based on the weighted utilization (load) of the NP_Ports connected to an FC fabric. Each new FLOGI or FDISC is assigned to the least-loaded link. On a link load rebalance, only the sessions that need to be moved to another link are logged out. When those sessions log in again, they are placed on active NP_Portal interfaces in a balanced manner.</td>
</tr>
<tr>
<td></td>
<td>• ENode-based—Load balancing is based on the ENode FLOGI. When an ENode logs in to the fabric, all subsequent FDISC sessions associated with that ENode are placed on the same link as the ENode FLOGI session, regardless of the link load. New ENode FLOGIs are placed on the least-loaded link. On a link load rebalance, all sessions are logged out. When the sessions log in again, they are placed on active NP_Portal interfaces in a balanced manner.</td>
</tr>
<tr>
<td></td>
<td>• FLOGI-based—Load balancing is based on the ENode FLOGI. When an ENode logs in to the fabric, all subsequent FDISC sessions associated with that ENode are placed on the same link as the ENode FLOGI session, regardless of the link load. New ENode FLOGIs are placed on the least-loaded link. On a link load rebalance, only the sessions that need to be moved to another link are logged out. When those sessions log in again, they are placed on active NP_Portal interfaces in a balanced manner.</td>
</tr>
</tbody>
</table>
Table 36: show fibre-channel proxy fabric-state Output Fields (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
</tr>
</thead>
</table>
| Fabric WWN verification | Fabric worldwide name (WWN) verification check state on the FCoE-FC gateway fabric:  
• Yes—Fabric WWN verification check is enabled.  
• No—Fabric WWN verification check is disabled. |
| Auto load rebalance enabled | Automated link load rebalancing configuration for the FCoE-FC gateway fabric:  
• No—Automated load balancing is disabled (default state).  
• Yes—Automated load balancing is enabled. |
| Last rebalance start-time | Time that the last link load rebalance began on the FCoE-FC gateway fabric:  
• Never—The link load has never been rebalanced.  
• Timestamp value—Time the last link load rebalancing started. |
| Last rebalance end-time   | Time that the last link load rebalance ended on the FCoE-FC gateway fabric:  
• Never—The link load has never been rebalanced.  
• Timestamp value—Time the last link load rebalancing ended. |
| Last rebalance trigger   | Event that triggered the last link load rebalance on the FCoE-FC gateway fabric:  
• None—The link load has never been rebalanced.  
• Config-CLI—Configure (enable) automated load balancing.  
• Request-CLI—Rebalance requested from the CLI using the request fibre-channel proxy load-rebalance fabric fabric-name operational command.  
• Preview-CLI—Rebalancing dry run requested from the CLI using the request fibre-channel proxy load-rebalance dry-run fabric fabric-name operational command. Indicates that the switch completed the dry run. A dry run simulates a link load rebalance and displays a list of sessions that might be affected if you request an actual rebalance.  
• Link-up—New FC link (NP_Port) up on the FCoE-FC gateway fabric, which causes a rebalance to distribute sessions to the new link.  
• Restore-complete—If the FC process on the switch restarts, the switch attempts to restore the session state that existed before the restart. When automated rebalance is enabled, restore-complete indicates that the sessions have been restored and rebalanced. |
| Last rebalance trigger-time | Time that the last link load rebalance was triggered on the FCoE-FC gateway fabric:  
• Never—Link load rebalancing has never been triggered.  
• Timestamp value—Time the last link load rebalancing was triggered. |
<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Last rebalance</td>
<td>Result of the last trigger event on the FCoE-FC gateway fabric:</td>
</tr>
<tr>
<td>trigger-result</td>
<td>• Never—Link load rebalancing has never been triggered.</td>
</tr>
<tr>
<td></td>
<td>• Not-configured—Automated rebalancing is not configured on the FCoE-FC gateway fabric.</td>
</tr>
<tr>
<td></td>
<td>• Not-required—Last rebalance trigger did not require rebalancing the link load (the link load was already balanced across the active NP_Port links).</td>
</tr>
<tr>
<td></td>
<td>• In-progress—Link load rebalancing is in progress and has not finished yet.</td>
</tr>
<tr>
<td></td>
<td>• Restore-in-progress—The switch is recovering from an FC process restart and is in the process of restoring the sessions to the active NP_Port links.</td>
</tr>
<tr>
<td></td>
<td>• Success—Link load rebalancing was successful.</td>
</tr>
<tr>
<td></td>
<td>• Logged-out-all—All sessions have been logged out.</td>
</tr>
<tr>
<td></td>
<td>• Preview-complete—The switch has finished simulating a dry run rebalancing request from the CLI (request fibre-channel proxy load-rebalance dry-run fabric [fabric-name] operational command) and reported the sessions that might be affected if you request an actual link load rebalance.</td>
</tr>
<tr>
<td></td>
<td>• Fabric-deletion-in-progress—FCoE-FC gateway fabric is in the process of being deleted.</td>
</tr>
</tbody>
</table>

**NOTE:** A trigger event does not necessarily result in a rebalance action. Link load rebalancing only occurs if the NP_Port interface session load is not balanced at the time of the trigger event.

Sample Output

show fibre-channel proxy fabric-state

```
user@switch> show fibre-channel proxy fabric-state
Fabric: san_fab1, Fabric-id: 10
Proxy load balance algorithm: Simple, Fabric WWN verification: Yes
Auto load rebalance enabled : No
Last rebalance start-time : Never
Last rebalance end-time : Never
Last rebalance trigger : Link-up
Last rebalance trigger-time : Mon Sep 10 21:42:30 2012 usec: 814602
Last rebalance trigger-result: Not-configured

Fabric: san_fab2, Fabric-id: 20
Proxy load balance algorithm: ENode based, Fabric WWN verification: Yes
Auto load rebalance enabled : No
Last rebalance start-time : Never
Last rebalance end-time : Never
Last rebalance trigger : Link-up
Last rebalance trigger-time : Mon Sep 17 17:23:35 2012 usec: 619684
Last rebalance trigger-result: Not-configured
```

show fibre-channel proxy fabric-state fabric

```
user@switch> show fibre-channel proxy fabric-state fabric fc_fabric_100
Fabric: fc_fabric_100, Fabric-id: 100
Proxy load balance algorithm: FLOGI based, Fabric WWN verification: No
Auto load rebalance enabled : Yes
Last rebalance start-time : Never
Last rebalance end-time : Never
Last rebalance trigger : Config-CLI
Last rebalance trigger-time : Fri Nov 2 08:56:16 2012 usec: 004487
Last rebalance trigger-result: Not-required
```
**show fibre-channel proxy login-table**

**Syntax**

```
show fibre-channel proxy login-table
  <brief | detail>
  (<fabric fabric-name>)
  (<interface interface-name>)
```

**Release Information**

Command introduced in Junos OS Release 11.1 for the QFX Series.

**Description**

Display Fibre Channel (FC) proxy fabric login table information.

**Options**

- `brief | detail` — (Optional) Display the specified level of output.
- `<interface interface-name>` — (Optional) Display output only for the specified interface.

**Required Privilege Level**

`view`

**Related Documentation**

- Configuring an FCoE-FC Gateway Fibre Channel Fabric
- show fibre-channel proxy fabric-state on page 313
- show fibre-channel proxy np-port on page 320
- show fibre-channel proxy statistics on page 323

**List of Sample Output**

- show fibre-channel proxy login-table on page 318
- show fibre-channel proxy login-table detail on page 318

**Output Fields**

Table 37 on page 317 lists the output fields for the `show fibre-channel proxy login-table` command. Output fields are listed in the approximate order in which they appear.

**Table 37: show fibre-channel proxy login-table Output Fields**

<table>
<thead>
<tr>
<th>Field Name (Type)</th>
<th>Field Description</th>
<th>Level of Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fabric</td>
<td>Name of the fabric.</td>
<td>All</td>
</tr>
<tr>
<td>Fabric-id</td>
<td>Fabric ID number.</td>
<td>All</td>
</tr>
<tr>
<td>F-Port</td>
<td>One of the following two values:</td>
<td>All</td>
</tr>
<tr>
<td></td>
<td>- VF_Port interface connected to the Fibre Channel over Ethernet (FCoE) host, shown as the FCoE VLAN interface.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- QFX Series FC port that is logged in to the FC switch, shown by a hyphen (-) to indicate that it is not the FCoE device VN_Port.</td>
<td></td>
</tr>
<tr>
<td>FCID</td>
<td>VN_Port Fibre Channel identifier provided by the Fibre Channel over Ethernet (FCoE) forwarder (FCF) or the Fibre Channel switch.</td>
<td>All</td>
</tr>
</tbody>
</table>
Table 37: show fibre-channel proxy login-table Output Fields (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
<th>Level of Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Port-WWN</td>
<td>Unique worldwide name (WWN) of the VN_Ports.</td>
<td>All</td>
</tr>
<tr>
<td>Node-WWN</td>
<td>Unique WWN of the node hosting the VN_Ports.</td>
<td>detail</td>
</tr>
<tr>
<td>NP-Port</td>
<td>NP_Portal interface connected to the FCoE forwarder (FCF) or the Fibre Channel switch.</td>
<td>All</td>
</tr>
<tr>
<td>Class</td>
<td>FLOGI service class.</td>
<td>detail</td>
</tr>
<tr>
<td>Fabric port WWN</td>
<td>Unique WWN of the fabric port (VF_Port).</td>
<td>detail</td>
</tr>
<tr>
<td>Fabric WWN</td>
<td>Unique WWN of the fabric generated by the FCF.</td>
<td>detail</td>
</tr>
</tbody>
</table>

Sample Output

show fibre-channel proxy login-table

```
user@switch> show fibre-channel proxy login-table
Fabric: proxy2, Fabric-id: 200
F-Port          FCID     Port-WWN               NP-Port
---             -------     --------                   -------
          -        0x030000 10:00:00:15:17:a9:98:64 fc-0/0/0.0
vlan.100       0x030001 20:00:10:94:00:01:00:01 fc-0/0/0.0
vlan.100       0x030002 20:00:10:94:00:01:00:05 fc-0/0/0.0
vlan.100       0x030003 20:00:10:94:00:02:00:01 fc-0/0/0.0
vlan.100       0x030004 20:00:10:94:00:02:00:05 fc-0/0/0.0
```

show fibre-channel proxy login-table detail

```
user@switch> show fibre-channel proxy login-table detail
Fabric: proxy2, Fabric-id: 200

FCID:             0x030000
F-Port:           wlan.100
NP-Port:          fc-0/0/0.0
Port WWN:         10:00:00:15:17:a9:98:64
Class:            3
Fabric port WWN:  10:00:00:15:17:a9:99:48
Fabric WWN:       00:0a:df:ff:0b:11:22:34

FCID:             0x030001
F-Port:           wlan.100
NP-Port:          fc-0/0/0.0
Port WWN:         20:00:10:94:00:01:00:01
Node WWN:         10:00:10:94:00:00:00:01
Class:            3
Fabric port WWN:  10:00:00:15:17:a9:99:48
Fabric WWN:       00:0a:df:ff:0b:11:22:34

FCID:             0x030002
F-Port:           wlan.100
NP-Port:          fc-0/0/0.0
Port WWN:         20:00:10:94:00:01:00:05
```
Node WWN: 10:00:10:94:00:00:00:01
Class: 3
Fabric port WWN: 10:00:00:15:17:a9:99:48
Fabric WWN: 00:0a:df:ff:0b:11:22:34
FCID: 0x030003
F-Port: vlan.100
NP-Port: fc-0/0/0.0
Port WWN: 20:00:10:94:00:02:00:01
Node WWN: 10:00:10:94:00:00:00:02
Class: 3
Fabric port WWN: 10:00:00:15:17:a9:99:48
Fabric WWN: 00:0a:df:ff:0b:11:22:34
FCID: 0x030004
F-Port: vlan.100
NP-Port: fc-0/0/0.0
Port WWN: 20:00:10:94:00:02:00:05
Node WWN: 10:00:10:94:00:00:00:02
Class: 3
Fabric port WWN: 10:00:00:15:17:a9:99:48
Fabric WWN: 00:0a:df:ff:0b:11:22:34
show fibre-channel proxy np-port

Syntax

show fibre-channel proxy np-port
  <brief | detail>
  <fabric fabric-name>
  <interface interface-name>

Release Information

Command introduced in Junos OS Release 11.1 for the QFX Series.

Description

Display Fibre Channel gateway fabric proxy Node Port (NP_Port) information.

Options

brief | detail—(Optional) Display the specified level of output.


interface interface-name—(Optional) Display output only for the specified interface.

Required Privilege

view

Related Documentation

- Configuring an FCoE-FC Gateway Fibre Channel Fabric
- Monitoring Fibre Channel Interface Load Balancing
  - show fibre-channel proxy fabric-state on page 313
  - show fibre-channel proxy login-table on page 317
  - show fibre-channel proxy statistics on page 323

List of Sample Output

show fibre-channel proxy np-port on page 321
show fibre-channel proxy np-port detail on page 321

Output Fields

Table 38 on page 320 lists the output fields for the show fibre-channel proxy np-port command. Output fields are listed in the approximate order in which they appear.

Table 38: show fibre-channel proxy np-port Output Fields

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
<th>Level of Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fabric</td>
<td>Name of the fabric.</td>
<td>All</td>
</tr>
<tr>
<td>Fabric-id</td>
<td>Fabric ID number.</td>
<td>All</td>
</tr>
<tr>
<td>NP-Port</td>
<td>NP_Port interface connected to the FCoE forwarder (FCF) or the Fibre Channel switch.</td>
<td>All</td>
</tr>
<tr>
<td>State</td>
<td>FCID state of the NP_Port interface.</td>
<td>All</td>
</tr>
<tr>
<td>Sessions</td>
<td>Number of active sessions on the NP_Port interface. A session is a FLOGI or FDISC login to the FC SAN fabric. Session does not refer to end-to-end storage sessions.</td>
<td>All</td>
</tr>
</tbody>
</table>
Table 38: show fibre-channel proxy np-port Output Fields (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
<th>Level of Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configured max login sessions</td>
<td>Configured maximum number of FIP login sessions permitted on the interface.</td>
<td>detail</td>
</tr>
<tr>
<td>Enodes</td>
<td>Number of ENodes with sessions on the NP_Port.</td>
<td>detail</td>
</tr>
<tr>
<td>LB state</td>
<td>Load-balancing state:</td>
<td>All</td>
</tr>
<tr>
<td></td>
<td>• On—Load balancing is on</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Off—Load balancing is off.</td>
<td></td>
</tr>
<tr>
<td>LB weight</td>
<td>Load balance weight, which reflects the port speed:</td>
<td>All</td>
</tr>
<tr>
<td></td>
<td>• 2—Port speed is 2 Gbps.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• 4—Port speed is 4 Gbps.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• 8—Port speed is 8 Gbps.</td>
<td></td>
</tr>
<tr>
<td>Ref-count</td>
<td>Reference count internal to Junos OS.</td>
<td>detail</td>
</tr>
<tr>
<td>Flags</td>
<td>Flags internal to Junos OS.</td>
<td>detail</td>
</tr>
</tbody>
</table>

NOTE: When an NP_Port interface reaches its configured maximum number of FIP sessions, the Flags field displays the flag MAX-LOGINS-REACHED.

Sample Output

show fibre-channel proxy np-port

```
user@switch> show fibre-channel proxy np-port
Fabric: proxy1, Fabric-id: 10
NP-Port | State  | Sessions | LB state | LB weight |
fc-0/0/0.0 | online | 3        | ON       | 4         |
fc-0/0/1.0 | online | 3        | ON       | 4         |
fc-0/0/2.0 | online | 3        | ON       | 4         |
root@junos1> show fibre-channel proxy np-port detail
```

show fibre-channel proxy np-port detail

```
user@switch> show fibre-channel proxy login-table detail
Fabric: proxy1, Fabric-id: 10
NP-Port: fc-0/0/0.0
State: online
Sessions: 3
Configured max login sessions: 130
Enodes: 1
LB state: ON
LB weight: 4
Ref-count: 4
Flags: UP LB
```
<table>
<thead>
<tr>
<th>NP-Port:</th>
<th>fc-0/0/1.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>State:</td>
<td>online</td>
</tr>
<tr>
<td>Sessions:</td>
<td>3</td>
</tr>
<tr>
<td>Configured max login sessions:</td>
<td>130</td>
</tr>
<tr>
<td>Enodes:</td>
<td>2</td>
</tr>
<tr>
<td>LB state:</td>
<td>ON</td>
</tr>
<tr>
<td>LB weight:</td>
<td>4</td>
</tr>
<tr>
<td>Ref-count:</td>
<td>4</td>
</tr>
<tr>
<td>Flags:</td>
<td>UP LB</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>NP-Port:</th>
<th>fc-0/0/2.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>State:</td>
<td>online</td>
</tr>
<tr>
<td>Sessions:</td>
<td>130</td>
</tr>
<tr>
<td>Configured max login sessions:</td>
<td>130</td>
</tr>
<tr>
<td>Enodes:</td>
<td>17</td>
</tr>
<tr>
<td>LB state:</td>
<td>OFF</td>
</tr>
<tr>
<td>LB weight:</td>
<td>4</td>
</tr>
<tr>
<td>Ref-count:</td>
<td>131</td>
</tr>
<tr>
<td>Flags:</td>
<td>UP MAX-LOGINS-REACHED</td>
</tr>
</tbody>
</table>
show fibre-channel proxy statistics

**Syntax**
```
show fibre-channel proxy statistics
  <fabric fabric-name>
```

**Release Information**
Command introduced in Junos OS Release 11.1 for the QFX Series.

**Description**
Display Fibre Channel proxy fabric statistics.

**Options**

**Required Privilege Level**
view

**Related Documentation**
- Configuring an FCoE-FC Gateway Fibre Channel Fabric
- show fibre-channel proxy fabric-state on page 313
- show fibre-channel proxy login-table on page 317
- show fibre-channel proxy np-port on page 320
- clear fibre-channel proxy statistics on page 232

**List of Sample Output**
show fibre-channel proxy statistics on page 324

**Output Fields**
Table 39 on page 323 lists the output fields for the show fibre-channel proxy statistics command. Output fields are listed in the approximate order in which they appear.

**Table 39: show fibre-channel proxy statistics Output Fields**

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fabric</td>
<td>Name of the fabric.</td>
</tr>
<tr>
<td>Fabric-id</td>
<td>Fabric ID number.</td>
</tr>
</tbody>
</table>
Table 39: show fibre-channel proxy statistics Output Fields (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NP-Port Transmit Command Statistics</td>
<td>Transmitted command statistics for the NP_Port.</td>
</tr>
<tr>
<td>• Command</td>
<td>Type of command issued on the NP_Port:</td>
</tr>
<tr>
<td>• FLOGI</td>
<td>Fabric login commands issued.</td>
</tr>
<tr>
<td>• FDISC</td>
<td>Fabric discovery commands issued.</td>
</tr>
<tr>
<td>• LOGO</td>
<td>Logout commands issued.</td>
</tr>
<tr>
<td>• Others</td>
<td>Other commands issued.</td>
</tr>
<tr>
<td>• Tx</td>
<td>Number of times the command type was transmitted.</td>
</tr>
<tr>
<td>• Rx-ACC</td>
<td>Number of times the NP_Port transmitted a receive accept message for the command type.</td>
</tr>
<tr>
<td>• Rx-RJT</td>
<td>Number of times the NP_Port transmitted a receive reject message for the command type.</td>
</tr>
<tr>
<td>• Abort</td>
<td>Number of times the NP_Port transmitted an abort message for the command type.</td>
</tr>
</tbody>
</table>

NP-Port Receive Command Statistics | Received command statistics for the NP_Port. |
| • Command | The type of command received on the NP_Port: |
| • LOGO | Logout commands issued. |
| • Others | Other commands issued. |
| • Rx | Number of times the command type was received. |
| • Tx-ACC | Number of times the NP_Port received a transmit accept message for the command type. |
| • Tx-RJT | Number of times the NP_Port received a transmit reject message for the command type. |
| • Abort | Number of times the NP_Port received an abort message for the command type. |

Sample Output
show fibre-channel proxy statistics

user@switch> show fibre-channel proxy statistics
Fabric: proxy1, Fabric-id: 10

<table>
<thead>
<tr>
<th>Command</th>
<th>Tx</th>
<th>Rx-ACC</th>
<th>Rx-RJT</th>
<th>Abort</th>
</tr>
</thead>
<tbody>
<tr>
<td>FLOGI</td>
<td>3</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>FDISC</td>
<td>3</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>LOGO</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Command</td>
<td>Rx</td>
<td>Tx-ACC</td>
<td>Tx-RJT</td>
<td>Abort</td>
</tr>
<tr>
<td>---------</td>
<td>----</td>
<td>--------</td>
<td>--------</td>
<td>-------</td>
</tr>
<tr>
<td>LOGO</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Others</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
show fip snooping

Syntax

```
show fip snooping
<brief | detail>
```

Release Information

Command introduced in Junos OS Release 10.4 for EX Series switches.
Command introduced in Junos OS Release 11.1 for the QFX Series.

Description

Display FIP snooping information.

Options

- **none**—Display FIP snooping information.
- **brief | detail**—(Optional) Display the specified level of output.

Required Privilege

- View

Related Documentation

- Configuring VN2VF_Port FIP Snooping and FCoE Trusted Interfaces on an FCoE Transit Switch on page 179
- Configuring an FCoE LAG
- Example: Configuring an FCoE Transit Switch
- Example: Configuring an FCoE LAG on a Redundant Server Node Group
- show fip snooping enode on page 331
- show fip snooping fcf on page 335
- show fip snooping interface on page 338
- show fip snooping statistics on page 341
- show fip snooping vlan on page 344

List of Sample Output

- show fip snooping on page 328
- show fip snooping brief (QFX Series) on page 328
- show fip snooping detail (QFX Series Switches) on page 329
- show fip snooping detail (QFabric System FCoE with LAG Configured) on page 329
- show fip snooping detail (EX Series Switches) on page 330

Output Fields

Table 40 on page 326 lists the output fields for the `show fip snooping` command. Output fields are listed in the approximate order in which they appear.

Table 40: show fip snooping Output Fields

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
<th>Level of Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>VLAN</td>
<td>Name of the VLAN.</td>
<td>All</td>
</tr>
</tbody>
</table>
Table 40: show fip snooping Output Fields (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
<th>Level of Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mode</td>
<td>(QFX Series only) Snooping mode enabled on the FCoE VLAN:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• VN2VF Snooping—The FCoE VLAN is configured for FIP snooping between an ENode VN_Port and a switch VF_Port.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• VN2VN Snooping—The FCoE VLAN is configured for VN_Port to VN_Port FIP snooping between ENode VN_Ports.</td>
<td></td>
</tr>
<tr>
<td>FC-MAP</td>
<td>FCoE mapped address prefix of the FCoE forwarder for the VLAN.</td>
<td>All</td>
</tr>
<tr>
<td>FCF or FCF-MAC</td>
<td>MAC address of the FCF.</td>
<td>All</td>
</tr>
<tr>
<td>Session Count or Active Sessions</td>
<td>Current number of virtual link sessions with VN_Ports.</td>
<td>All</td>
</tr>
<tr>
<td>VN_Port Count</td>
<td>(QFX Series only) Number of VN_Ports active on an ENode.</td>
<td>brief</td>
</tr>
<tr>
<td>Configured FKA-ADV</td>
<td>FIP keepalive interval in seconds configured on the FCF multiplied by three. For example, if the FKA_ADV period configured on the FCF is 86 seconds, the value of this field is 258. For the QFX Series only, the output of this field is always 0 (zero) if the VLAN is an FCoE-FC gateway VLAN. If the VLAN is a FIP snooping VLAN (a transit switch VLAN), then the output is accurate. This is because for an FCoE-FC gateway VLAN, FIP snooping is performed internally and the keepalive advertisements are not tracked by the switch's Ethernet module.</td>
<td>detail</td>
</tr>
<tr>
<td>Running FKA-ADV</td>
<td>Runtime interval in seconds of the last FIP keepalive advertisement the FCF received. This value changes every time the FCF receives an FKA_ADV. For the QFX Series only, the output of this field is always 0 (zero) if the VLAN is an FCoE-FC gateway VLAN. If the VLAN is a FIP snooping VLAN (a transit switch VLAN), then the output is accurate. This is because for an FCoE-FC gateway VLAN, FIP snooping is performed internally and the keepalive advertisements are not tracked by the switch's Ethernet module.</td>
<td>detail</td>
</tr>
<tr>
<td>Beacon Period</td>
<td>(QFX Series only) Beacon period interval in milliseconds.</td>
<td>detail</td>
</tr>
</tbody>
</table>
Table 40: show fip snooping Output Fields (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
<th>Level of Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>VN2VN Mode</td>
<td>(QFX Series only) Mode of VN2VN_Port snooping:</td>
<td>detail</td>
</tr>
<tr>
<td></td>
<td>• Multi-Point—Multiple ENodes are connected to the network and form multiple virtual links. Each virtual link is created between one pair of VN_Ports. This is analogous to the loop mode in traditional FC networks.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Point-to-Point—Two ENodes are connected to the network and form a single VN_Port to VN_Port virtual link. This is analogous to the point-to-point FC link between an FC initiator and an FC target.</td>
<td></td>
</tr>
<tr>
<td>ENode-MAC</td>
<td>MAC address of the connected FCoE node (ENode).</td>
<td>All</td>
</tr>
<tr>
<td>Interface</td>
<td>Interface connected to the ENode.</td>
<td>detail</td>
</tr>
<tr>
<td></td>
<td>(QFabric System only) When an FCoE LAG has been configured, LAG interface connected to the ENode and LAG member interface connected to ENode.</td>
<td></td>
</tr>
<tr>
<td>VN-Port MAC</td>
<td>MAC address of a VN_Port on the ENode.</td>
<td>All</td>
</tr>
<tr>
<td>FKA-ADV</td>
<td>Runtime interval in seconds of the last FIP keepalive advertisement the ENode sent to the FCF on behalf of the VN_Port (VN_Port FKA_ADV). This value changes every time the ENode sends a VN_Port FKA_ADV to the FCF.</td>
<td>detail</td>
</tr>
<tr>
<td>Active VN_Ports</td>
<td>(QFX Series only) Number of VN_Ports active on an ENode.</td>
<td>detail</td>
</tr>
<tr>
<td>Vlink far-end VN-Port-MAC</td>
<td>Media access control (MAC) address of the VN_Port at the other end of the virtual link.</td>
<td>detail</td>
</tr>
</tbody>
</table>

Sample Output
show fip snooping

```
user@switch> show fip snooping
VLAN : fcoevlan1  FC-MAP : 0e:fc:00
FCF : 00:10:94:00:00:01 Session Count : 2
Enode-MAC : 00:10:94:00:00:02
VN-Port-MAC : 0e:fc:00:01:00:01
VN-Port-MAC : 0e:fc:00:01:00:05
```

show fip snooping brief (QFX Series)

```
user@switch> show fip snooping brief
VLAN: vlan100,   Mode: VN2VF Snooping
FC-MAP: 0e:fc:00
FCF: 30:10:94:01:00:00 Session Count: 2
Enode-MAC: 10:10:94:01:00:01
```
show fip snooping detail (QFX Series Switches)

user@switch> show fip snooping detail
root@sw-pa02v> show fip snooping detail
VLAN: vlan100, Mode: VN2VF Snooping
FC-MAP: 0e:fc:00
FCF Information
FCF-MAC            : 30:10:94:01:00:00
Active Sessions    : 2
Configured FKA-ADV : 258
Running FKA-ADV    : 188
Enode Information
Enode-MAC: 10:10:94:01:00:01, Interface: xe-0/0/10
Configured FKA-ADV : 258
Running FKA-ADV    : 230
Session Information
VN-Port MAC: 0e:fc:00:01:0d:01, FKA-ADV : 230
VN-Port MAC: 0e:fc:00:01:0e:01, FKA-ADV : 245
VLAN: vlan101, Mode: VN2VN Snooping
FC-MAP: 0e:fd:00
Beacon_Period:  90000
VN2VN Mode: Multi-Point
Enode Information
Enode-MAC: 10:10:94:01:00:02, Interface: xe-0/0/10
Active VN_Ports : 1
VN_Port Information
VN-Port MAC: 0e:fd:00:01:0a:01
Active Sessions : 2
Session Information
Vlink far-end VN-Port-MAC: 0e:fd:00:01:0b:01
Vlink far-end VN-Port-MAC: 0e:fd:00:01:0c:01
Enode-MAC: 10:10:94:01:00:01, Interface: xe-0/0/11
Active VN_Ports : 0

show fip snooping detail (QFabric System FCoE with LAG Configured)

admin@qfabric> show fip snooping detail
VLAN: vlan_100, Mode: VN2VF Snooping
FC-MAP: 0e:fc:00
FCF Information
FCF-MAC            : 84:18:88:d1:f5:cc
Active Sessions    : 2
Configured FKA-ADV : 8000
Running FKA-ADV    : 23962
Enode Information
Enode-MAC: 00:c0:dd:14:ae:6d, Interface: P4546-C:ae0  P4546-C:xe-0/0/39
Configured FKA-ADV : 8000
Running FKA-ADV    : 16622
Session Information
VN-Port MAC: 0e:fc:00:6c:06:a5, FKA-ADV : 246303
Enode Information
Enode-MAC: 00:c0:dd:14:ae:6f, Interface: P4546-C:ae0 P4546-C:xe-0/0/38

Configured FKA-ADV : 8000
Running FKA-ADV : 16512
Session Information
VN-Port MAC: 0e:fc:00:6c:06:a4, FKA-ADV : 238150

show fip snooping detail (EX Series Switches)

user@switch> show fip snooping detail
VLAN : fcoevlan1 FC-MAP : 0e:fc:00
FCF Information
FCF-MAC : 00:10:94:00:00:01
Active Sessions : 2
Configured FKA-ADV : 258
Running FKA-ADV : 244

Enode Information
Enode-MAC : 00:10:94:00:00:02 Interface : xe-0/0/1
Configured FKA-ADV : 258
Running FKA-ADV : 244
Session Information
VN-Port MAC : 0E:FC:00:01:00:05 FKA-ADV : 264
VN-Port MAC : 0E:FC:00:01:00:01 FKA-ADV : 260
show fip snooping enode

Syntax

`show fip snooping enode enode-mac
  <brief | detail>
  <vlan vlan-name>`

Release Information

Command introduced in Junos OS Release 10.4 for EX Series switches.
Command introduced in Junos OS Release 11.1 for the QFX Series.

Description

Display FIP snooping FCoE node (ENode) information.

Options

`brief | detail`—(Optional) Display the specified level of output.

`enode-mac`—Display information for the ENode specified by the MAC address.

`vlan vlan-name`—(Optional) Display FIP snooping information for the ENode on only the specified VLAN.

Required Privilege

`view`

Level

Related Documentation

- Configuring VN2VF_Port FIP Snooping and FCoE Trusted Interfaces on an FCoE Transit Switch on page 179
- **Example: Configuring an FCoE Transit Switch**
  - show fip snooping on page 326
  - show fip snooping fcf on page 335
  - show fip snooping interface on page 338
  - show fip snooping statistics on page 341
  - show fip snooping vlan on page 344

List of Sample Output

- show fip snooping enode on page 333
- show fip snooping enode brief (QFX Series) on page 333
- show fip snooping enode detail (QFX Series) on page 333
- show fip snooping enode detail on page 333

Output Fields

Table 41 on page 331 lists the output fields for the **show fip snooping enode** command. Output fields are listed in the approximate order in which they appear.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
<th>Level of Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENode and ENode MAC</td>
<td>MAC address of the ENode.</td>
<td>All</td>
</tr>
<tr>
<td>VLAN</td>
<td>Name of the VLAN.</td>
<td>All</td>
</tr>
<tr>
<td>Interface</td>
<td>Interface connected to the ENode.</td>
<td>All</td>
</tr>
</tbody>
</table>
Table 41: show fip snooping enode Output Fields (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
<th>Level of Output</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Mode</strong></td>
<td>(QFX Series only) Snooping mode enabled on the FCoE VLAN:</td>
<td>All</td>
</tr>
<tr>
<td></td>
<td>- VN2VF Snooping—The FCoE VLAN is configured for FIP snooping between an ENode VN_Port and a switch VF_Port.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- VN2VN Snooping—The FCoE VLAN is configured for VN_Port to VN_Port FIP snooping between ENode VN_Ports.</td>
<td></td>
</tr>
<tr>
<td>VN_Port Count</td>
<td>(QFX Series only) Number of VN_Ports active on an ENode.</td>
<td>brief</td>
</tr>
<tr>
<td>Session Count</td>
<td>Current number of virtual link sessions with VN_Ports.</td>
<td>All</td>
</tr>
<tr>
<td>Configured FKA-ADV</td>
<td>FIP keepalive interval in seconds configured on the FCoE forwarder (FCF) multiplied by three. For example, if the FKA ADV period configured on the FCF is 86 seconds, the value of this field is 258. This value remains constant. For the QFX Series only, the output of this field is always 0 (zero) if the VLAN is an FCoE-FC gateway VLAN. If the VLAN is a FIP snooping VLAN (a transit switch VLAN), then the output is accurate. This is because for an FCoE-FC gateway VLAN, FIP snooping is performed internally and the keepalive advertisements are not tracked by the switch’s Ethernet module.</td>
<td>detail</td>
</tr>
<tr>
<td>Running FKA-ADV</td>
<td>Runtime interval in seconds of the last FIP keepalive advertisement the ENode sent to the FCF. This value changes every time the ENode sends an FKAADV to the FCF. For the QFX Series only, the output of this field is always 0 (zero) if the VLAN is an FCoE-FC gateway VLAN. If the VLAN is a FIP snooping VLAN (a transit switch VLAN), then the output is accurate. This is because for an FCoE-FC gateway VLAN, FIP snooping is performed internally and the keepalive advertisements are not tracked by the switch’s Ethernet module.</td>
<td>detail</td>
</tr>
<tr>
<td>VN-Port or VN-Port-MAC</td>
<td>MAC address of a VN_Port on the ENode.</td>
<td>All</td>
</tr>
<tr>
<td>FKA-ADV</td>
<td>Runtime interval in seconds of the last FIP keepalive advertisement the ENode sent to the FCF on behalf of the VN_Port (VN_Port FKA-ADV). This value changes every time the ENode sends a VN_Port FKA-ADV to the FCF.</td>
<td>detail</td>
</tr>
<tr>
<td>FCF or FCF-MAC</td>
<td>MAC address of the FCF to which the VN_Port is connected.</td>
<td>All</td>
</tr>
<tr>
<td>Beacon Period</td>
<td>(QFX Series only) Beacon period interval in milliseconds.</td>
<td>detail</td>
</tr>
</tbody>
</table>
Table 41: show fip snooping enode Output Fields (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
<th>Level of Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>VN2VN Mode</td>
<td>(QFX Series only) Mode of VN2VN_Port snooping:</td>
<td>detail</td>
</tr>
<tr>
<td></td>
<td>• Multi-Point—Multiple ENodes are connected to the network and form multiple</td>
<td></td>
</tr>
<tr>
<td></td>
<td>virtual links. Each virtual link is created between one pair of VN_Ports. This</td>
<td></td>
</tr>
<tr>
<td></td>
<td>is analogous to the loop mode in traditional FC networks.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Point-to-Point—Two ENodes are connected to the network and form a single VN_Port</td>
<td></td>
</tr>
<tr>
<td></td>
<td>to VN_Port virtual link. This is analogous to the point-to-point FC link between</td>
<td></td>
</tr>
<tr>
<td></td>
<td>an FC initiator and an FC target.</td>
<td></td>
</tr>
<tr>
<td>Vlink far-end VN-Port-MAC</td>
<td>(QFX Series only) Media access control (MAC) address of the VN_Port at the other</td>
<td>detail</td>
</tr>
<tr>
<td></td>
<td>end of the virtual link.</td>
<td></td>
</tr>
</tbody>
</table>

Sample Output

show fip snooping enode

user@switch> show fip snooping enode 00:10:94:00:00:02
Enode : 00:10:94:00:00:02   VLAN : vlan1   Interface : xe-0/0/1
VN-Port-MAC     FCF-MAC
0E:FC:00:00:00:05     00:10:94:00:00:01
0E:FC:00:00:00:01     00:10:94:00:00:01

show fip snooping enode brief (QFX Series)

user@switch> show fip snooping enode 10:10:94:01:00:02 brief
Enode: 10:10:94:01:00:02 , VLAN: vlan101, Interface: xe-0/0/10
Mode: VN2VF Snooping      VN_Port Count: 1
VN_Port Information
VN_Port Mac: 0e:fc:00:01:0a:01      Session Count: 2

show fip snooping enode detail (QFX Series)

user@switch> show fip snooping enode 10:10:94:01:00:02 detail
Enode MAC: 10:10:94:01:00:02:02, VLAN: vlan101, Interface: xe-0/0/10
Mode: VN2VF Snooping      VN_Port Count: 1
Beacon_Period: 90000      VN2VN Mode: Multi-Point
VN_Port Information
VN_Port Mac: 0e:fc:00:01:0a:01      Session Count: 2
Vlink far-end VN-Port-MAC: 0e:fc:00:01:0b:01
Vlink far-end VN-Port-MAC: 0e:fc:00:01:0c:01

show fip snooping enode detail

user@switch> show fip snooping enode 00:10:94:00:00:02 detail
Enode MAC : 00:10:94:00:00:02   VLAN : vlan1   Interface : xe-0/0/1
Configured FKA-ADV : 258   Running FKA-ADV : 213
Session Information
VN-Port : 0E:FC:00:00:00:05   FKA-ADV : 229   FCF : 00:10:94:00:00:01
VN-Port : 0E:FC:00:00:00:01   FKA-ADV : 225   FCF : 00:10:94:00:00:01
show fip snooping fcf

Syntax

show fip snooping fcf fcf-mac
  <brief | detail>
  <vlan vlan-name>

Release Information

Command introduced in Junos OS Release 10.4 for EX Series switches.
Command introduced in Junos OS Release 11.1 for the QFX Series.

Description

Display FIP snooping FCoE forwarder (FCF) information.

Options

brief | detail—(Optional) Display the specified level of output.

fcf-mac—Display information for the FCF specified by the MAC address.

vlan-name—(Optional) Display FIP snooping information for the FCF on only the specified VLAN.

Required Privilege

view

Related Documentation

• Configuring VN2VF_Port FIP Snooping and FCoE Trusted Interfaces on an FCoE Transit Switch on page 179
  • Example: Configuring an FCoE Transit Switch
  • show fip snooping on page 326
  • show fip snooping enode on page 331
  • show fip snooping interface on page 338
  • show fip snooping statistics on page 341
  • show fip snooping vlan on page 344

List of Sample Output

show fip snooping fcf on page 336
show fip snooping fcf detail on page 336

Output Fields

Table 42 on page 335 lists the output fields for the show fip snooping fcf command. Output fields are listed in the approximate order in which they appear.

Table 42: show fip snooping fcf Output Fields

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
<th>Level of Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>FCF or FCF-MAC</td>
<td>MAC address of the FCoE forwarder.</td>
<td>All</td>
</tr>
<tr>
<td>VLAN</td>
<td>Name of the VLAN.</td>
<td>All</td>
</tr>
<tr>
<td>Session Count</td>
<td>Current number of virtual link sessions with VN_Ports.</td>
<td>None</td>
</tr>
</tbody>
</table>
Table 42: show fip snooping fcf Output Fields (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
<th>Level of Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configured FKA-ADV</td>
<td>FIP keepalive interval in seconds configured on the FCF multiplied by three. For example, if the FKA_ADV period configured on the FCF is 86 seconds, the value of this field is 258.</td>
<td>detail</td>
</tr>
<tr>
<td>Running FKA-ADV</td>
<td>Runtime interval in seconds of the last FIP keepalive advertisement the FCF received. This value changes every time the FCF receives an FKA_ADV.</td>
<td>detail</td>
</tr>
<tr>
<td>ENode-MAC</td>
<td>MAC address of the connected ENode.</td>
<td>All</td>
</tr>
<tr>
<td>• Interface</td>
<td>Interface connected to the ENode.</td>
<td>detail</td>
</tr>
<tr>
<td>• Configured FKA-ADV</td>
<td>FIP keepalive interval in seconds configured on the FCF multiplied by three. For example, if the FKA_ADV period configured on the FCF is 86 seconds, the value of this field is 258. This value remains constant.</td>
<td>detail</td>
</tr>
<tr>
<td>• Running FKA-ADV</td>
<td>Runtime interval in seconds of the last FIP keepalive advertisement the ENode sent to the FCF. This value changes every time the ENode sends an FKA_ADV to the FCF.</td>
<td>detail</td>
</tr>
<tr>
<td>• VN-Port MAC</td>
<td>MAC address of a VN_Port on the ENode.</td>
<td>All</td>
</tr>
<tr>
<td>• FKA-ADV</td>
<td>Runtime interval in seconds of the last FIP keepalive advertisement the ENode sent to the FCF on behalf of the VN_Port (VN_Port FKA_ADV). This value changes every time the ENode sends a VN_Port FKA_ADV to the FCF.</td>
<td>detail</td>
</tr>
</tbody>
</table>

Sample Output
show fip snooping fcf

user@switch> show fip snooping fcf 00:10:94:00:00:01
FCF : 00:10:94:00:00:01  VLAN : vlan1  Session Count : 2
Enode-MAC : 00:10:94:00:00:02
VN-Port-MAC : 0E:FC:00:00:00:05
VN-Port-MAC : 0E:FC:00:00:00:01

show fip snooping fcf detail

user@switch> show fip snooping fcf 00:10:94:00:00:01 detail
FCF-MAC : 00:10:94:00:00:01  VLAN : vlan1  Configured FKA-ADV : 258  Running FKA-ADV : 222
Enode Information
Enode-MAC : 00:10:94:00:00:02  Interface: xe-0/0/1
Configured FKA-ADV : 258
Running FKA-ADV : 226
Session Information
VN-Port MAC : 0E:FC:00:00:00:05  FKA-ADV : 242
VN-Port MAC : 0E:FC:00:00:00:01  FKA-ADV : 238
**show fip snooping interface**

**Syntax**
```
show fip snooping interface interface-name
  <brief | detail>
```

**Release Information**
Command introduced in Junos OS Release 12.1 for the QFX Series.

**Description**
Display FIP snooping information for the specified interface.

**Options**
- **brief** | **detail**—(Optional) Display the specified level of output.
- **interface-name**—Display information for the specified interface.

**Required Privilege Level**
view

**Related Documentation**
- Configuring VN2VF_Port FIP Snooping and FCoE Trusted Interfaces on an FCoE Transit Switch on page 179
- show fip snooping on page 326
- show fip snooping enode on page 331
- show fip snooping fcf on page 335
- show fip snooping statistics on page 341
- show fip snooping vlan on page 344

**List of Sample Output**
- show fip snooping interface on page 340
- show fip snooping interface detail on page 340

**Output Fields**
Table 43 on page 338 lists the output fields for the `show fip snooping interface interface-name` command. Output fields are listed in the approximate order in which they appear.

### Table 43: show fip snooping interface Output Fields

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
<th>Level of Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>VLAN</td>
<td>Name of the VLAN.</td>
<td>All</td>
</tr>
<tr>
<td>FC-MAP</td>
<td>FCoE mapped address prefix of the FCoE forwarder for the VLAN.</td>
<td>All</td>
</tr>
<tr>
<td>FCF or FCF-MAC</td>
<td>MAC address of the FCF.</td>
<td>All</td>
</tr>
<tr>
<td>Session Count or Active Sessions</td>
<td>Current number of virtual link sessions with VN_Ports.</td>
<td>All</td>
</tr>
</tbody>
</table>
Table 43: show fip snooping interface Output Fields (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
<th>Level of Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configured FKA-ADV</td>
<td>FIP keepalive interval in seconds configured on the FCF multiplied by three. For example, if the FKA_ADV period configured on the FCF is 86 seconds, the value of this field is 258. For the QFX Series only, the output of this field is always 0 (zero) if the VLAN is an FCoE-FC gateway VLAN. If the VLAN is a FIP snooping VLAN (a transit switch VLAN), then the output is accurate. This is because for an FCoE-FC gateway VLAN, FIP snooping is performed internally and the keepalive advertisements are not tracked by the switch's Ethernet module.</td>
<td>detail</td>
</tr>
<tr>
<td>Running FKA-ADV</td>
<td>Runtime interval in seconds of the last FIP keepalive advertisement the FCF received. This value changes every time the FCF receives an FKA_ADV. For the QFX Series only, the output of this field is always 0 (zero) if the VLAN is an FCoE-FC gateway VLAN. If the VLAN is a FIP snooping VLAN (a transit switch VLAN), then the output is accurate. This is because for an FCoE-FC gateway VLAN, FIP snooping is performed internally and the keepalive advertisements are not tracked by the switch's Ethernet module.</td>
<td>detail</td>
</tr>
<tr>
<td>ENode-MAC</td>
<td>MAC address of the connected FCoE node (ENode).</td>
<td>All</td>
</tr>
<tr>
<td>Interface</td>
<td>Interface connected to the ENode.</td>
<td>detail</td>
</tr>
<tr>
<td>Configured FKA-ADV</td>
<td>FIP keepalive interval in seconds configured on the FCF multiplied by three. For example, if the FKA_ADV period configured on the FCF is 86 seconds, the value of this field is 258. This value remains constant. For the QFX Series only, the output of this field is always 0 (zero) if the VLAN is an FCoE-FC gateway VLAN. If the VLAN is a FIP snooping VLAN (a transit switch VLAN), then the output is accurate. This is because for an FCoE-FC gateway VLAN, FIP snooping is performed internally and the keepalive advertisements are not tracked by the switch's Ethernet module.</td>
<td>detail</td>
</tr>
<tr>
<td>Running FKA-ADV</td>
<td>Runtime interval in seconds of the last FIP keepalive advertisement the ENode sent to the FCF. This value changes every time the ENode sends an FKA_ADV to the FCF. For the QFX Series only, the output of this field is always 0 (zero) if the VLAN is an FCoE-FC gateway VLAN. If the VLAN is a FIP snooping VLAN (a transit switch VLAN), then the output is accurate. This is because for an FCoE-FC gateway VLAN, FIP snooping is performed internally and the keepalive advertisements are not tracked by the switch's Ethernet module.</td>
<td>detail</td>
</tr>
<tr>
<td>VN-Port MAC</td>
<td>MAC address of a VN_Port on the ENode.</td>
<td>All</td>
</tr>
</tbody>
</table>
### Table 43: show fip snooping interface Output Fields (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
<th>Level of Output</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FKA-ADV</strong></td>
<td>Runtime interval in seconds of the last FIP keepalive advertisement the ENode sent to the FCF on behalf of the VN_Port (VN_Port FKA_ADV). This value changes every time the ENode sends a VN_Port FKA_ADV to the FCF.</td>
<td><strong>detail</strong></td>
</tr>
</tbody>
</table>

### Sample Output

**show fip snooping interface**

```bash
user@switch> show fip snooping interface xe-0/0/9.0
VLAN: vlan_100, FC-MAP: 0e:fc:00
    FCF: 30:10:94:01:00:00 Session Count: 1
    Enode-MAC: 10:10:94:01:00:01
    VN-Port-MAC: 0e:fc:00:01:0a:01
```

**show fip snooping interface detail**

```bash
user@switch> show fip snooping interface xe-0/0/9.0 detail
VLAN: vlan_100, FC-MAP: 0e:fc:00
    FCF Information
        FCF-MAC : 30:10:94:01:00:00
        Active Sessions : 1
        Configured FKA-ADV : 368640000
        Running FKA-ADV : 0
    Enode Information
        Enode-MAC: 10:10:94:01:00:01, Interface: xe-0/0/9
        Configured FKA-ADV : 368640000
        Running FKA-ADV : 0
    Session Information
        VN-Port MAC: 0e:fc:00:01:0a:01, FKA-ADV : 0
```
**show fip snooping statistics**

**Syntax**  
show fip snooping statistics  
<vlan vlan-name>

**Release Information**  
Command introduced in Junos OS Release 10.4 for EX Series switches. Command introduced in Junos OS Release 11.1 for the QFX Series.

**Description**  
Display FIP snooping statistics.

**Options**  
<vlan vlan-name>—(Optional) Display FIP snooping statistics for the specified VLAN.

**Required Privilege Level**  
view

**Related Documentation**
- Example: Configuring an FCoE Transit Switch
- Configuring VN2VF_Port FIP Snooping and FCoE Trusted Interfaces on an FCoE Transit Switch on page 179
- show fip snooping on page 326
- show fip snooping enode on page 331
- show fip snooping fcf on page 335
- show fip snooping interface on page 338
- show fip snooping vlan on page 344

**List of Sample Output**
- show fip snooping statistics (FIP Snooping) on page 343
- show fip snooping statistics (VN2VN_Port Snooping) on page 343

**Output Fields**  
Table 44 on page 341 lists the output fields for the **show fip snooping statistics** command. Output fields are listed in the approximate order in which they appear.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>VLAN</td>
<td>Name of the VLAN for which a set of statistics is displayed.</td>
</tr>
</tbody>
</table>
| Mode       | (QFX Series only) Snooping mode enabled on the FCoE VLAN:  
  - VN2VF Snooping—The FCoE VLAN is configured for FIP snooping between an ENode VN_Port and a switch VF_Port.  
  - VN2VN Snooping—The FCoE VLAN is configured for VN_Port to VN_Port FIP snooping between ENode VN_Ports. |
| Number of MDS | Number of multicast discovery solicitation messages sent on the VLAN. |
Table 44: show fip snooping statistics Output Fields (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of UDS</td>
<td>Number of unicast discovery solicitation messages sent on the VLAN.</td>
</tr>
<tr>
<td>Number of FLOGI</td>
<td>Number of fabric logins on the VLAN.</td>
</tr>
<tr>
<td>Number of FDISC</td>
<td>Number of fabric discovery logins on the VLAN.</td>
</tr>
<tr>
<td>Number of LOGO</td>
<td>Number of fabric logouts on the VLAN.</td>
</tr>
<tr>
<td>Number of ENode-keep-alive</td>
<td>Number of ENode keepalive messages sent on the VLAN.</td>
</tr>
<tr>
<td>Number of VNPort-keep-alive</td>
<td>Number of VN_Port keepalive messages sent on the VLAN.</td>
</tr>
<tr>
<td>Number of MDA</td>
<td>Number of multicast discovery advertisement messages sent on the VLAN.</td>
</tr>
<tr>
<td>Number of UDA</td>
<td>Number of unicast discovery advertisement messages sent on the VLAN.</td>
</tr>
<tr>
<td>Number of FLOGI_ACC</td>
<td>Number of fabric logins accepted on the VLAN.</td>
</tr>
<tr>
<td>Number of FLOGI_RJT</td>
<td>Number of fabric logins rejected on the VLAN.</td>
</tr>
<tr>
<td>Number of FDISC_ACC</td>
<td>Number of fabric discoveries accepted on the VLAN.</td>
</tr>
<tr>
<td>Number of FDISC_RJT</td>
<td>Number of fabric discoveries rejected on the VLAN.</td>
</tr>
<tr>
<td>Number of LOGO_ACC</td>
<td>Number of fabric logouts accepted on the VLAN.</td>
</tr>
<tr>
<td>Number of LOGO_RJT</td>
<td>Number of fabric logouts rejected on the VLAN.</td>
</tr>
<tr>
<td>Number of CVL</td>
<td>Number of clear virtual links (CVL) actions on the VLAN.</td>
</tr>
<tr>
<td>Number of VN_Port Probes Req</td>
<td>Number of multicast N_Port_ID probes sent to the ALL-VN2VN-ENode-MACs multicast address on the VLAN.</td>
</tr>
<tr>
<td>Number of VN_Port Claim Notif</td>
<td>Number of multicast N_Port_ID claim notifications sent on the VLAN.</td>
</tr>
<tr>
<td>Number of VN_Port Beacons</td>
<td>Number of multicast beacons sent on the VLAN.</td>
</tr>
<tr>
<td>Number of VN_Port Probes Reply</td>
<td>Number of replies to N_Port_ID probes sent on the VLAN. Replies are unicast to the ENode MAC address of the probe requester.</td>
</tr>
</tbody>
</table>
Table 44: show fip snooping statistics Output Fields (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of VN_Port Claim Reply</td>
<td>(QFX Series only) Number of replies to N_Port_ID claim notifications sent on the VLAN. Replies are unicast to the ENode MAC address of the claim notifier.</td>
</tr>
</tbody>
</table>

Sample Output

show fip snooping statistics (FIP Snooping)

```
user@switch>  show fip snooping statistics
VLAN: fcoevlan1   Mode: VN2VF Snooping
  Number of MDS:               2
  Number of UDS:               2
  Number of FLOGI:             2
  Number of FDISC:             2
  Number of LOGO:              0
  Number of Enode-keep-alive:  200
  Number of VNPort-keep-alive: 200
  Number of MDA:               25
  Number of UDA:               2
  Number of FLOGI_ACC:         2
  Number of FLOGI_RJT:         0
  Number of FDISC_ACC:         2
  Number of FDISC_RJT:         0
  Number of LOGO_ACC:          0
  Number of LOGO_RJT:          0
  Number of CVL:               0
```

show fip snooping statistics (VN2VN_Port Snooping)

```
user@switch>  show fip snooping statistics
VLAN: vlan101   Mode: VN2VN Snooping
  Number of VN_Port Probes Req:        3
  Number of VN_Port Claim Notif:       3
  Number of VN_Port Beacons:           0
  Number of VN_Port Probes Reply:      3
  Number of VN_Port Claim Reply:       3
  Number of FLOGI:                      0
  Number of FLOGI_ACC:                  0
  Number of FLOGI_RJT:                  0
  Number of FDISC:                      0
  Number of FDISC_ACC:                  0
  Number of FDISC_RJT:                  0
  Number of LOGO:                       0
  Number of LOGO_ACC:                   0
  Number of LOGO_RJT:                   0
```
### show fip snooping vlan

**Syntax**

```
show fip snooping vlan vlan-name <brief | detail>
```

**Release Information**

Command introduced in Junos OS Release 10.4 for EX Series switches.
Command introduced in Junos OS Release 11.1 for the QFX Series.

**Description**

Display FIP snooping VLAN information.

**Options**

- `brief | detail`—(Optional) Display the specified level of output.
- `vlan-name`—Display information for the specified VLAN.

**Required Privilege Level**

view

**Related Documentation**

- Configuring VN2VF_Port FIP Snooping and FCoE Trusted Interfaces on an FCoE Transit Switch on page 179
- *Example: Configuring an FCoE Transit Switch*
  - show fip snooping on page 326
  - show fip snooping enode on page 331
  - show fip snooping fcf on page 335
  - show fip snooping interface on page 338
  - show fip snooping statistics on page 341

**List of Sample Output**

- show fip snooping vlan on page 346
- show fip snooping vlan (QFX Series, VN2VF_Port FIP Snooping) on page 346
- show fip snooping vlan (QFX Series, VN2VN_Port FIP Snooping) on page 346
- show fip snooping vlan detail (QFX Series, VN2VN_Port FIP Snooping) on page 347
- show fip snooping vlan detail on page 347

**Output Fields**

Table 45 on page 344 lists the output fields for the `show fip snooping vlan` command. Output fields are listed in the approximate order in which they appear.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
<th>Level of Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>VLAN</td>
<td>Name of the VLAN.</td>
<td>All</td>
</tr>
</tbody>
</table>
Table 45: show fip snooping vlan Output Fields (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
<th>Level of Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mode</td>
<td>(QFX Series only) Snooping mode enabled on the FCoE VLAN:</td>
<td>All</td>
</tr>
<tr>
<td></td>
<td>• VN2VF Snooping—The FCoE VLAN is configured for FIP snooping between an ENode VN_Port and a switch VF_Port.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• VN2VN Snooping—The FCoE VLAN is configured for VN_Port to VN_Port FIP snooping between ENode VN_Ports.</td>
<td></td>
</tr>
<tr>
<td>VN_Port count</td>
<td>(QFX Series only) Number of VN_Ports active on an ENode when the mode is VN2VN_Port FIP snooping.</td>
<td>All</td>
</tr>
<tr>
<td>FC-MAP</td>
<td>FCoE mapped address prefix of the FCoE forwarder for the VLAN.</td>
<td>All</td>
</tr>
<tr>
<td>Beacon_Period</td>
<td>(QFX Series only) Beacon period interval in milliseconds.</td>
<td>detail</td>
</tr>
<tr>
<td>VN2VN Mode</td>
<td>(QFX Series only) Mode of VN2VN_Port snooping:</td>
<td>detail</td>
</tr>
<tr>
<td></td>
<td>• Multi-Point—Multiple ENodes are connected to the network and form multiple virtual links. Each virtual link is created between one pair of VN_Ports. This is analogous to the loop mode in traditional FC networks.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Point-to-Point—Two ENodes are connected to the network and form a single VN_Port to VN_Port virtual link. This is analogous to the point-to-point FC link between an FC initiator and an FC target.</td>
<td></td>
</tr>
<tr>
<td>FCF or FCF-MAC</td>
<td>MAC address of the FCF.</td>
<td>All</td>
</tr>
<tr>
<td>Session Count or Active Sessions</td>
<td>Current number of virtual link sessions with VN_Ports.</td>
<td>All</td>
</tr>
<tr>
<td>Configured FKA-ADV</td>
<td>FIP keepalive interval in seconds configured on the FCF multiplied by three. For example, if the FKA_ADV period configured on the FCF is 86 seconds, the value of this field is 258.</td>
<td>detail</td>
</tr>
<tr>
<td>Running FKA-ADV</td>
<td>Runtime interval in seconds of the last FIP keepalive advertisement the FCF received. This value changes every time the FCF receives an FKA_ADV.</td>
<td>detail</td>
</tr>
</tbody>
</table>
Table 45: show fip snooping vlan Output Fields (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
<th>Level of Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENode-MAC</td>
<td>MAC address of the connected ENode.</td>
<td>All</td>
</tr>
<tr>
<td>• Interface</td>
<td>Interface connected to the ENode.</td>
<td>detail</td>
</tr>
<tr>
<td>• Configured FKA-ADV</td>
<td>FIP keepalive interval in seconds configured on the FCF multiplied by three. For example, if the FKA_ADV period configured on the FCF is 86 seconds, the value of this field is 258. This value remains constant.</td>
<td>detail</td>
</tr>
<tr>
<td>• Running FKA-ADV</td>
<td>Runtime interval in seconds of the last FIP keepalive advertisement the ENode sent to the FCF. This value changes every time the ENode sends an FKA_ADV to the FCF.</td>
<td>detail</td>
</tr>
<tr>
<td>• VN-Port MAC</td>
<td>MAC address of a VN_Port on the ENode.</td>
<td>All</td>
</tr>
<tr>
<td>• FKA-ADV</td>
<td>Runtime interval in seconds of the last FIP keepalive advertisement the ENode sent to the FCF on behalf of the VN_Port (VN_Port FKA_ADV). This value changes every time the ENode sends a VN_Port FKA_ADV to the FCF.</td>
<td>detail</td>
</tr>
<tr>
<td>• Active VN_Ports</td>
<td>(QFX Series only) Number of VN_Ports active on an ENode.</td>
<td>detail</td>
</tr>
<tr>
<td>• Vlink far-end VN-Port-MAC</td>
<td>(QFX Series only) Media access control (MAC) address of the VN_Port at the other end of the virtual link.</td>
<td>detail</td>
</tr>
</tbody>
</table>

Sample Output

show fip snooping vlan

```
user@switch> show fip snooping vlan fcoevlan1
VLAN : fcoevlan1    FC-MAP : 0e:fc:00
    FCF : 00:10:94:00:00:01   Session Count : 2
    ENode-MAC : 00:10:94:00:00:02
    VN-Port-MAC : 0E:FC:00:00:00:05
    VN-Port-MAC : 0E:FC:00:00:00:01
```

show fip snooping vlan (QFX Series, VN2VF_Port FIP Snooping)

```
user@switch> show fip snooping vlan fcoevlan1
VLAN : fcoevlan1    Mode: VN2VF Snooping
    FC-MAP : 0e:fc:00
    FCF : 00:10:94:00:00:01   Session Count : 2
    ENode-MAC : 00:10:94:00:00:02
    VN-Port-MAC : 0E:FC:00:00:00:05
    VN-Port-MAC : 0E:FC:00:00:00:01
```

show fip snooping vlan (QFX Series, VN2VN_Port FIP Snooping)

```
user@switch> show fip snooping vlan vlan101
```
VLAN: vlan101,  Mode: VN2VN Snooping
   FC-MAP: 0e:fd:00
   Enode-MAC: 10:10:94:01:00:02  VN_Port count: 1
   VN-Port-MAC: 0e:fd:00:00:0a:01  Session Count: 2
   Enode-MAC: 10:10:94:01:00:03  VN_Port count: 0

show fip snooping vlan detail (QFX Series, VN2VN_Port FIP Snooping)

user@switch> show fip snooping vlan vlan101 detail
VLAN: vlan101,  Mode: VN2VN Snooping
   FC-MAP: 0e:fd:00
   Beacon_Period: 90000
   VN2VN Mode: Multi-Point
   Enode Information
   Enode-MAC: 10:10:94:01:00:02,       Interface: xe-0/0/10
   Active VN_Ports : 1
   VN_Port Information
   VN-Port MAC: 0e:fd:00:00:0a:01
   Active Sessions : 2
   Session Information
   Vlink far-end VN-Port-MAC: 0e:fd:00:00:0b:01
   Vlink far-end VN-Port-MAC: 0e:fd:00:00:0c:01
   Enode-MAC: 10:10:94:01:00:02,       Interface: xe-0/0/11
   Active VN_Ports : 0

show fip snooping vlan detail

user@switch> show fip snooping vlan fcoevlan1 detail
VLAN : fcoevlan1    FC-MAP : 0e:fc:00
   FCF Information
   FCF-MAC            : 00:10:94:00:00:01
   Active Sessions    : 2
   Configured FKA-ADV : 258
   Running FKA-ADV    : 235
   Enode Information
   Enode-MAC : 00:10:94:00:00:02       Interface : xe-0/0/1
   Configured FKA-ADV : 258
   Running FKA-ADV    : 239
   Session Information
   VN-Port MAC : 0E:FC:00:00:00:05   FKA-ADV : 255
   VN-Port MAC : 0E:FC:00:00:00:01   FKA-ADV : 251
**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**
- clear fip vlan-discovery statistics on page 236
- Understanding FIP Functions on page 32
- Understanding FIP Implementation on an FCoE-FC Gateway

**List of Sample Output**
- show fip vlan-discovery enodes on page 349
- show fip vlan-discovery statistics (QFX3500) on page 349
- show fip vlan-discovery statistics (QFabric Systems) on page 349

**Output Fields**
Table 46 on page 348 lists the output fields for the show fip vlan-discovery command. Output fields are listed in the approximate order in which they appear.

**Table 46: show fip vlan-discovery Output Fields**

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
<th>Level of Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENode-MAC</td>
<td>Media access control (MAC) address of the ENode.</td>
<td>enodes</td>
</tr>
<tr>
<td>Interface</td>
<td>Name of the interface.</td>
<td>enodes</td>
</tr>
<tr>
<td>Unsolicited notification count</td>
<td>Number of unsolicited VLAN discovery notifications.</td>
<td>All</td>
</tr>
<tr>
<td>Solicited notification count</td>
<td>Number of solicited VLAN discovery notifications.</td>
<td>statistics</td>
</tr>
<tr>
<td>Node Group Name</td>
<td>Displays the name of the Node group on QFabric systems.</td>
<td>statistics</td>
</tr>
<tr>
<td>Request count</td>
<td>Number of VLAN discovery requests sent by the ENode. This number should match the Solicited notification count number.</td>
<td>statistics</td>
</tr>
<tr>
<td>VLAN tags</td>
<td>Tags of the FIP-enabled VLANs.</td>
<td>enodes</td>
</tr>
</tbody>
</table>
**Sample Output**

**show fip vlan-discovery enodes**

```
user@switch> show fip vlan-discovery enodes

<table>
<thead>
<tr>
<th>Enode-MAC</th>
<th>Interface</th>
<th>Unsolicited Notification Count</th>
<th>Vlan Tags</th>
</tr>
</thead>
<tbody>
<tr>
<td>00:10:94:00:00:02</td>
<td>xe-0/0/9.0</td>
<td>0</td>
<td>400</td>
</tr>
</tbody>
</table>
```

**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
```
show route forwarding-table family fibre-channel

Syntax
show route forwarding-table family fibre-channel
  <brief | detail | extensive>
  <all>
  <destination destination-prefix>
  <interface-name interface-name>
  <label label>
  <matching ip-prefix>
  <multicast>
  <summary>
  <table routing-table-name>
  <vlan vlan-name>
  <vpn vpn-instance-name>

Release Information
Command introduced in Junos OS Release 11.1 for the QFX Series.

Description
Display Fibre Channel family forwarding table route information.

Options
  brief | detail | extensive—(Optional) Display the specified level of output.
  all—Display all routing forwarding tables.
  destination destination-prefix—Destination prefix.
  interface-name interface-name—Name of the interface.
  label label—Display route entries for the specified label name.
  matching ip-prefix—Display route entries for the specified IP prefix or length.
  multicast—Display multicast routes.
  summary—Display route count instead of details.
  table routing-table-name—Name of the routing table.
  vlan vlan-name—Name of the VLAN.
  vpn vpn-instance-name—Name of the VPN instance.

Required Privilege
  Level
  view

Related Documentation
  • show fibre-channel next-hops on page 309
  • show fibre-channel routes on page 311

List of Sample Output
  show route forwarding-table family fibre-channel on page 351

Output Fields
Table 47 on page 351 lists the output fields for the show route forwarding-table family fibre-channel command. Output fields are listed in the approximate order in which they appear.
Table 47: show route forwarding-table family fibre-channel Output Fields

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Routing table</td>
<td>Name of the routing table.</td>
</tr>
<tr>
<td>Destination</td>
<td>Route destination.</td>
</tr>
<tr>
<td>Type</td>
<td>Type of route internal to Junos OS.</td>
</tr>
<tr>
<td>RtRef</td>
<td>Route reference count internal to Junos OS.</td>
</tr>
<tr>
<td>Next hop Type</td>
<td>Type of next hop internal to Junos OS.</td>
</tr>
<tr>
<td>Index</td>
<td>Next-hop index identifier.</td>
</tr>
<tr>
<td>NhRef</td>
<td>Number of routes that refer to the next hop.</td>
</tr>
<tr>
<td>Netif</td>
<td>Interface used to reach the next hop.</td>
</tr>
</tbody>
</table>

**Sample Output**

show route forwarding-table family fibre-channel

```
user@switch> show route forwarding-table family fibre-channel
Routing table: default.fibre-channel
Fibre Channel:
  Destination       Type   RtRef  Next hop          Type  Index  NhRef  Netif
  default           perm   0      dscd               126   1
  0x30000/24        user    0      ucst               674   2  fc-0/0/0.0
  0x30001/24        user    0      ucst               675   2  vlan.100
  0x30002/24        user    0      ucst               676   2  vlan.100
  0x30003/24        user    0      ucst               677   2  vlan.100
  0x30004/24        user    0      ucst               678   2  vlan.100
```
PART 4

Troubleshooting

• Troubleshooting Procedures on page 355
CHAPTER 11

Troubleshooting Procedures

- Troubleshooting Dropped FCoE Traffic on page 355
- Troubleshooting Dropped FIP Traffic on page 358

Troubleshooting Dropped FCoE Traffic

<table>
<thead>
<tr>
<th>Problem</th>
<th>Description: Fibre Channel over Ethernet (FCoE) traffic for which you want guaranteed delivery is dropped.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cause</td>
<td>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.):</td>
</tr>
</tbody>
</table>

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.

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

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.

Chapter 11: Troubleshooting Procedures
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 115 for step-by-step instructions on how to configure PFC for FCoE traffic, including classifier, interface, congestion notification profile, PFC, and bandwidth scheduling configuration.

---

**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
   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
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

   For example, to name the native VLAN native and use the VLAN ID 1:
   
   [edit]
   
   user@switch# set vlans native vlan-id

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

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

- Understanding FIP Functions on page 32
- Configuring VLANs for FCoE Traffic on an FCoE Transit Switch on page 174