



Junos[®] OS

CCC and TCC Feature Guide for Routing Devices

Release

14.1



Published: 2014-05-14

Juniper Networks, Inc.
1194 North Mathilda Avenue
Sunnyvale, California 94089
USA
408-745-2000
www.juniper.net

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

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

Junos[®] OS CCC and TCC Feature Guide for Routing Devices

14.1

Copyright © 2014, Juniper Networks, Inc.
All rights reserved.

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

YEAR 2000 NOTICE

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

END USER LICENSE AGREEMENT

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

Table of Contents

| | | |
|------------------|---|----------|
| | About the Documentation | ix |
| | Documentation and Release Notes | ix |
| | Supported Platforms | ix |
| | Using the Examples in This Manual | ix |
| | Merging a Full Example | x |
| | Merging a Snippet | x |
| | Documentation Conventions | xi |
| | Documentation Feedback | xiii |
| | Requesting Technical Support | xiii |
| | Self-Help Online Tools and Resources | xiii |
| | Opening a Case with JTAC | xiv |
| Part 1 | Overview | |
| Chapter 1 | Introduction to CCC and TCC | 3 |
| | CCC Overview | 3 |
| | Transmitting Nonstandard BPDUs | 4 |
| | TCC Overview | 4 |
| | CCC and TCC Graceful Restart | 5 |
| Part 2 | Configuration | |
| Chapter 2 | CCC and TCC Configuration Guidelines | 9 |
| | Configuring Layer 2 Switching Cross-Connects Using CCC | 9 |
| | Configuring the CCC Encapsulation for Layer 2 Switching | |
| | Cross-Connects | 10 |
| | Configuring ATM Encapsulation for Layer 2 Switching | |
| | Cross-Connects | 10 |
| | Configuring Ethernet Encapsulation for Layer 2 Switching | |
| | Cross-Connects | 11 |
| | Configuring Ethernet VLAN Encapsulation for Layer 2 Switching | |
| | Cross-Connects | 11 |
| | Configuring Aggregated Ethernet Encapsulation for Layer 2 Switching | |
| | Cross-Connects | 12 |
| | Configuring Frame Relay Encapsulation for Layer 2 Switching | |
| | Cross-Connects | 13 |
| | Configuring PPP and Cisco HDLC Encapsulation for Layer 2 Switching | |
| | Cross-Connects | 14 |
| | Configuring the CCC Connection for Layer 2 Switching Cross-Connects | 14 |
| | Configuring MPLS for Layer 2 Switching Cross-Connects | 14 |

| | | |
|------------------|--|-----------|
| | Example: Configuring a Layer 2 Switching Cross-Connect | 15 |
| | Configuring MPLS LSP Tunnel Cross-Connects Using CCC | 17 |
| | Configuring the CCC Encapsulation for LSP Tunnel Cross-Connects | 18 |
| | Configuring the CCC Connection for LSP Tunnel Cross-Connects | 19 |
| | Example: Configuring an LSP Tunnel Cross-Connect | 20 |
| | Configuring TCC | 21 |
| | Configuring the Encapsulation for Layer 2 Switching TCCs | 21 |
| | Configuring PPP and Cisco HDLC Encapsulation for Layer 2 Switching TCCs | 22 |
| | Configuring ATM Encapsulation for Layer 2 Switching TCCs | 22 |
| | Configuring Frame Relay Encapsulation for Layer 2 Switching TCCs | 22 |
| | Configuring Ethernet Encapsulation for Layer 2 Switching TCCs | 23 |
| | Configuring Ethernet Extended VLAN Encapsulation for Layer 2 Switching TCCs | 23 |
| | Configuring ARP for Ethernet and Ethernet Extended VLAN Encapsulations | 24 |
| | Configuring the Connection for Layer 2 Switching TCCs | 25 |
| | Configuring MPLS for Layer 2 Switching TCCs | 25 |
| | Configuring CCC and TCC Graceful Restart | 26 |
| | Configuring CCC Switching for Point-to-Multipoint LSPs | 26 |
| | Configuring the Point-to-Multipoint LSP Switch on Ingress PE Routers | 27 |
| | Configuring Local Receivers on a Point-to-Multipoint CCC LSP Switch on Ingress PE Routers | 27 |
| | Configuring the Point-to-Multipoint LSP Switch on Egress PE Routers | 28 |
| Chapter 3 | CCC and TCC Configuration Statements | 29 |
| | connections (Circuits) | 30 |
| | encapsulation (Logical Interface) | 31 |
| | encapsulation (Physical Interface) | 35 |
| | interface-switch | 40 |
| | output-interface (CCC) | 40 |
| | lsp-switch | 41 |
| | p2mp-receive-switch | 42 |
| | p2mp-transmit-switch | 43 |
| | remote-interface-switch | 44 |
| Part 3 | Administration | |
| Chapter 4 | CCC and TCC Operational Commands | 47 |
| | show connections | 48 |
| | show mpls lsp | 51 |
| | show route ccc | 67 |
| | show route forwarding-table | 68 |
| Part 4 | Index | |
| | Index | 85 |

List of Figures

| | | |
|------------------|---|----------|
| Part 1 | Overview | |
| Chapter 1 | Introduction to CCC and TCC | 3 |
| | Figure 1: TCC Example | 4 |
| | Figure 2: Remote Interface Switch Connecting Two CE Routers Using CCC | 5 |
| Part 2 | Configuration | |
| Chapter 2 | CCC and TCC Configuration Guidelines | 9 |
| | Figure 3: Layer 2 Switching Cross-Connect | 9 |
| | Figure 4: Topology of a Frame Relay Layer 2 Switching Cross-Connect | 15 |
| | Figure 5: Sample Topology of a VLAN Layer 2 Switching Cross-Connect | 16 |
| | Figure 6: MPLS Tunnel Cross-Connect | 17 |
| | Figure 7: Example Topology of MPLS LSP Tunnel Cross-Connect | 20 |

List of Tables

| | | |
|------------------|--|-----------|
| | About the Documentation | ix |
| | Table 1: Notice Icons | xi |
| | Table 2: Text and Syntax Conventions | xii |
| Part 3 | Administration | |
| Chapter 4 | CCC and TCC Operational Commands | 47 |
| | Table 3: show connections Output Fields | 49 |
| | Table 4: show mpls lsp Output Fields | 53 |
| | Table 5: show route forwarding-table Output Fields | 71 |

About the Documentation

- [Documentation and Release Notes on page ix](#)
- [Supported Platforms on page ix](#)
- [Using the Examples in This Manual on page ix](#)
- [Documentation Conventions on page xi](#)
- [Documentation Feedback on page xiii](#)
- [Requesting Technical Support on page xiii](#)

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:

- [MX Series](#)
- [T Series](#)
- [M Series](#)
- [PTX Series](#)

Using the Examples in This Manual

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

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

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

Merging a Full Example

To merge a full example, follow these steps:

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

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

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

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

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

Merging a Snippet

To merge a snippet, follow these steps:

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

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

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

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

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

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

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

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

Documentation Conventions

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

Table 1: Notice Icons







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

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

Table 2: Text and Syntax Conventions

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

GUI Conventions

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

| Convention | Description | Examples |
|------------------------------|--|---|
| Bold text like this | Represents graphical user interface (GUI) items you click or select. | <ul style="list-style-type: none"> In the Logical Interfaces box, select All Interfaces. To cancel the configuration, click Cancel. |
| > (bold right angle bracket) | Separates levels in a hierarchy of menu selections. | In the configuration editor hierarchy, select Protocols>Ospf . |

Documentation Feedback

We encourage you to provide feedback, comments, and suggestions so that we can improve the documentation. You can send your comments to techpubs-comments@juniper.net, or fill out the documentation feedback form at <https://www.juniper.net/cgi-bin/docbugreport/>. If you are using e-mail, be sure to include the following information with your comments:

- Document or topic name
- URL or page number
- Software release 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.

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

Self-Help Online Tools and Resources

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

- Find CSC offerings: <http://www.juniper.net/customers/support/>
- Search for known bugs: <http://www2.juniper.net/kb/>

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

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

Opening a Case with JTAC

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

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

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

PART 1

Overview

- [Introduction to CCC and TCC on page 3](#)

CHAPTER 1

Introduction to CCC and TCC

- [CCC Overview on page 3](#)
- [Transmitting Nonstandard BPDUs on page 4](#)
- [TCC Overview on page 4](#)
- [CCC and TCC Graceful Restart on page 5](#)

CCC Overview

Circuit cross-connect (CCC) allows you to configure transparent connections between two circuits, where a circuit can be a Frame Relay data-link connection identifier (DLCI), an Asynchronous Transfer Mode (ATM) virtual circuit (VC), a Point-to-Point Protocol (PPP) interface, a Cisco High-Level Data Link Control (HDLC) interface, or an MPLS label-switched path (LSP). Using CCC, packets from the source circuit are delivered to the destination circuit with, at most, the Layer 2 address being changed. No other processing—such as header checksums, time-to-live (TTL) decrementing, or protocol processing—is done.

CCC circuits fall into two categories: logical interfaces, which include DLCIs, VCs, virtual local area network (VLAN) IDs, PPP and Cisco HDLC interfaces, and LSPs. The two circuit categories provide three types of cross-connect:

- **Layer 2 switching**—Cross-connects between logical interfaces provide what is essentially Layer 2 switching. The interfaces that you connect must be of the same type.
- **MPLS tunneling**—Cross-connects between interfaces and LSPs allow you to connect two distant interface circuits of the same type by creating MPLS tunnels that use LSPs as the conduit.
- **LSP stitching**—Cross-connects between LSPs provide a way to “stitch” together two label-switched paths, including paths that fall in two different traffic engineering database areas.

For Layer 2 switching and MPLS tunneling, the cross-connect is bidirectional, so packets received on the first interface are transmitted out the second interface, and those received on the second interface are transmitted out the first. For LSP stitching, the cross-connect is unidirectional.

You can police (control) the amount of traffic flowing over CCC circuits. For more information, see the *Junos OS VPNs Library for Routing Devices*.

It is also possible to use the **ping** command to check the integrity of CCC LSPs. See *Pinging CCC LSPs* for more information.

Transmitting Nonstandard BPDUs

CCC protocol (and Layer 2 Circuit and Layer 2 VPN) configurations can transmit nonstandard bridge protocol data units (BPDUs) generated by other vendors' equipment. This is the default behavior on all supported PICs and requires no additional configuration.

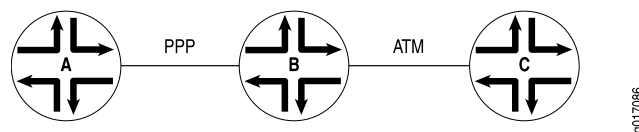
The following PICs are supported on M320 and T Series routers:

- 1-port Gigabit Ethernet PIC
- 2-port Gigabit Ethernet PIC
- 4-port Gigabit Ethernet PIC
- 10-port Gigabit Ethernet PIC

TCC Overview

Translational cross-connect (TCC) is a switching concept that enables you to establish interconnections between a variety of Layer 2 protocols or circuits. It is similar to CCC. However, whereas CCC requires the same Layer 2 encapsulations on each side of a Juniper Networks router (such as PPP-to-PPP or Frame Relay-to-Frame Relay), TCC enables you to connect different types of Layer 2 protocols interchangeably. When you use TCC, combinations such as PPP-to-ATM (see [Figure 1 on page 4](#)) and Ethernet-to-Frame Relay connections are possible.

Figure 1: TCC Example



The Layer 2 circuits and encapsulation types that can be interconnected by TCC are:

- Ethernet
- Extended VLANs
- PPP
- HDLC
- ATM
- Frame Relay

TCC works by removing the Layer 2 header when frames enter the router and adding a different Layer 2 header on the frames before they leave the router. In [Figure 1 on page 4](#), the PPP encapsulation is stripped from the frames arriving at Router B, and the ATM encapsulation is added before the frames are sent to Router C.

Note that all control traffic is terminated at the interconnecting router (Router B). Examples of traffic controllers include the Link Control Protocol (LCP) and the Network Control Protocol (NCP) for PPP, keepalives for HDLC, and Local Management Interface (LMI) for Frame Relay.

TCC functionality is different from standard Layer 2 switching. TCC only swaps Layer 2 headers. No other processing, such as header checksums, TTL decrementing, or protocol handling is performed. TCC is supported for IPv4 only.

Address Resolution Protocol (APR) packet policing on TCC Ethernet interfaces is effective for releases 10.4 and onwards.

You can configure TCC for interface switching and for Layer 2 VPNs. For more information about using TCC for virtual private networks (VPNs), see the *Junos OS VPNs Library for Routing Devices*.

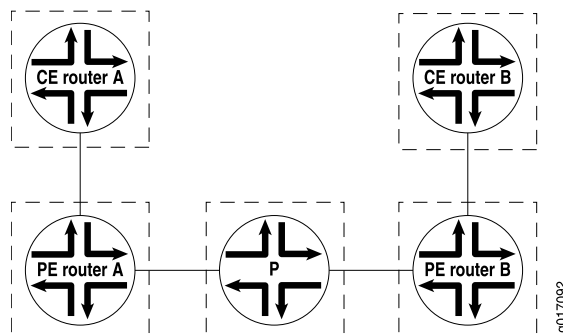
CCC and TCC Graceful Restart

CCC and TCC graceful restart allows Layer 2 connections between customer edge (CE) routers to restart gracefully. These Layer 2 connections are configured with the **remote-interface-switch** or **lsp-switch** statements. Because these CCC and TCC connections have an implicit dependency on RSVP LSPs, graceful restart for CCC and TCC uses the RSVP graceful restart capabilities.

RSVP graceful restart must be enabled on the PE routers and P routers to enable graceful restart for CCC and TCC. Also, because RSVP is used as the signaling protocol for signaling label information, the neighboring router must use helper mode to assist with the RSVP restart procedures.

Figure 2 on page 5 illustrates how graceful restart might work on a CCC connection between two CE routers.

Figure 2: Remote Interface Switch Connecting Two CE Routers Using CCC



PE Router A is the ingress for the transmit LSP from PE Router A to PE Router B and the egress for the receive LSP from PE Router B to PE Router A. With RSVP graceful restart enabled on all the PE and P routers, the following occurs when PE router A restarts:

- PE Router A preserves the forwarding state associated with the CCC routes (those from CCC to MPLS and from MPLS to CCC).

- Traffic flows without disruption from CE router to CE router.
- After the restart, PE Router A preserves the label for the LSP for which PE Router A is the egress (the receive LSP, for example). The transmit LSP from PE Router A to PE Router B can derive new label mappings, but should not cause any traffic disruption.

PART 2

Configuration

- [CCC and TCC Configuration Guidelines on page 9](#)
- [CCC and TCC Configuration Statements on page 29](#)

CHAPTER 2

CCC and TCC Configuration Guidelines

- [Configuring Layer 2 Switching Cross-Connects Using CCC on page 9](#)
- [Configuring MPLS LSP Tunnel Cross-Connects Using CCC on page 17](#)
- [Configuring TCC on page 21](#)
- [Configuring CCC and TCC Graceful Restart on page 26](#)
- [Configuring CCC Switching for Point-to-Multipoint LSPs on page 26](#)

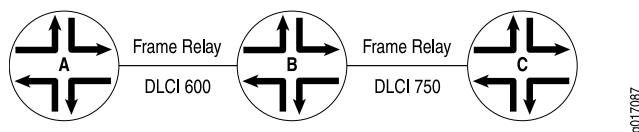
Configuring Layer 2 Switching Cross-Connects Using CCC

Layer 2 switching cross-connects join logical interfaces to form what is essentially Layer 2 switching. The interfaces that you connect must be of the same type.

[Figure 3 on page 9](#) illustrates a Layer 2 switching cross-connect. In this topology, Router A and Router C have Frame Relay connections to Router B, which is a Juniper Networks router. Circuit cross-connect (CCC) allows you to configure Router B to act as a Frame Relay (Layer 2) switch.

To configure Router B to act as a Frame Relay switch, you configure a circuit from Router A to Router C that passes through Router B, effectively configuring Router B as a Frame Relay switch with respect to these routers. This configuration allows Router B to transparently switch packets (frames) between Router A and Router C without regard to the packets' contents or the Layer 3 protocols. The only processing that Router B performs is to translate DLCI 600 to 750.

Figure 3: Layer 2 Switching Cross-Connect



If the Router A-to-Router B and Router B-to-Router C circuits were PPP, for example, the Link Control Protocol and Network Control Protocol exchanges occur between Router A and Router C. These messages are handled transparently by Router B, allowing Router A and Router C to use various PPP options (such as header or address compression and authentication) that Router B might not support. Similarly, Router A and Router C exchange keepalives, providing circuit-to-circuit connectivity status.

You can configure Layer 2 switching cross-connects on PPP, Cisco HDLC, Frame Relay, Ethernet, and ATM circuits. In a single cross-connect, only like interfaces can be connected.

To configure Layer 2 switching cross-connects, you must configure the following on the router that is acting as the switch (Router B in [Figure 3 on page 9](#)):

- [Configuring the CCC Encapsulation for Layer 2 Switching Cross-Connects on page 10](#)
- [Configuring the CCC Connection for Layer 2 Switching Cross-Connects on page 14](#)
- [Configuring MPLS for Layer 2 Switching Cross-Connects on page 14](#)
- [Example: Configuring a Layer 2 Switching Cross-Connect on page 15](#)

Configuring the CCC Encapsulation for Layer 2 Switching Cross-Connects

To configure Layer 2 switching cross-connects, configure the CCC encapsulation on the router that is acting as the switch (Router B in [Figure 3 on page 9](#)).



NOTE: You cannot configure families on CCC interfaces; that is, you cannot include the family statement at the [edit interfaces *interface-name* unit *logical-unit-number*] hierarchy level.

For instructions for configuring the encapsulation for Layer 2 switching cross-connects, see the following sections:

- [Configuring ATM Encapsulation for Layer 2 Switching Cross-Connects on page 10](#)
- [Configuring Ethernet Encapsulation for Layer 2 Switching Cross-Connects on page 11](#)
- [Configuring Ethernet VLAN Encapsulation for Layer 2 Switching Cross-Connects on page 11](#)
- [Configuring Aggregated Ethernet Encapsulation for Layer 2 Switching Cross-Connects on page 12](#)
- [Configuring Frame Relay Encapsulation for Layer 2 Switching Cross-Connects on page 13](#)
- [Configuring PPP and Cisco HDLC Encapsulation for Layer 2 Switching Cross-Connects on page 14](#)

Configuring ATM Encapsulation for Layer 2 Switching Cross-Connects

For ATM circuits, specify the encapsulation when configuring the virtual circuit (VC). Configure each VC as a circuit or a regular logical interface by including the following statements:

```
at-fpc/pic/port {  
  atm-options {  
    vpi vpi-identifier maximum-vcs maximum-vcs;  
  }  
  unit logical-unit-number {  
    encapsulation encapsulation-type;  
    point-to-point; # Default interface type  
    vci vpi-identifier.vci-identifier;
```

```
}
}
```

You can include these statements at the following hierarchy levels:

- [edit interfaces]
- [edit logical-systems *logical-system-name* interfaces]

Configuring Ethernet Encapsulation for Layer 2 Switching Cross-Connects

For Ethernet circuits, specify **ethernet-ccc** in the **encapsulation** statement. This statement configures the entire physical device. For these circuits to work, you must also configure a logical interface (unit 0).

Ethernet interfaces with standard Tag Protocol Identifier (TPID) tagging can use Ethernet CCC encapsulation. On M Series Multiservice Edge Routers, except the M320, one-port Gigabit Ethernet, two-port Gigabit Ethernet, four-port Gigabit Ethernet, and four-port Fast Ethernet PICs can use Ethernet CCC encapsulation. On T Series Core Routers and M320 routers, one-port Gigabit Ethernet and two-port Gigabit Ethernet PICs installed in FPC2 can use Ethernet CCC encapsulation. When you use this encapsulation type, you can configure the **ccc** family only.

```
fe-fpc/pic/port {
  encapsulation ethernet-ccc;
  unit 0;
}
```

You can include these statements at the following hierarchy levels:

- [edit interfaces]
- [edit logical-systems *logical-system-name* interfaces]

Configuring Ethernet VLAN Encapsulation for Layer 2 Switching Cross-Connects

An Ethernet virtual LAN (VLAN) circuit can be configured using either the **vlan-ccc** or **extended-vlan-ccc** encapsulation. If you configure the **extended-vlan-ccc** encapsulation on the physical interface, you cannot configure the **inet** family on the logical interfaces. Only the **ccc** family is allowed. If you configure the **vlan-ccc** encapsulation on the physical interface, both the **inet** and **ccc** families are supported on the logical interfaces. Ethernet interfaces in VLAN mode can have multiple logical interfaces.

For encapsulation type **vlan-ccc**, VLAN IDs from 512 through 4094 are reserved for CCC VLANs. For the **extended-vlan-ccc** encapsulation type, all VLAN IDs 1 and higher are valid. VLAN ID 0 is reserved for tagging the priority of frames.



NOTE: Some vendors use the proprietary TPIDs 0x9100 and 0x9901 to encapsulate a VLAN-tagged packet into a VLAN-CCC tunnel to interconnect a geographically separated metro Ethernet network. By configuring the **extended-vlan-ccc** encapsulation type, a Juniper Networks router can accept all three TPIDs (0x8100, 0x9100, and 0x9901).

Configure an Ethernet VLAN circuit with the **vlan-ccc** encapsulation as follows:

```
interfaces {  
  type-fpc/pic/port {  
    vlan-tagging;  
    encapsulation vlan-ccc;  
    unit logical-unit-number {  
      encapsulation vlan-ccc;  
      vlan-id vlan-id;  
    }  
  }  
}
```

You can configure these statements at the following hierarchy levels:

- **[edit interfaces]**
- **[edit logical-systems *logical-system-name* interfaces]**

Configure an Ethernet VLAN circuit with the **extended-vlan-ccc** encapsulation statement as follows:

```
interfaces {  
  type-fpc/pic/port {  
    vlan-tagging;  
    encapsulation extended-vlan-ccc;  
    unit logical-unit-number {  
      vlan-id vlan-id;  
      family ccc;  
    }  
  }  
}
```

You can configure these statements at the following hierarchy levels:

- **[edit interfaces]**
- **[edit logical-systems *logical-system-name* interfaces]**

Whether you configure the encapsulation as **vlan-ccc** or **extended-vlan-ccc**, you must enable VLAN tagging by including the **vlan-tagging** statement.

Configuring Aggregated Ethernet Encapsulation for Layer 2 Switching Cross-Connects

You can configure aggregated Ethernet interfaces for CCC connections and for Layer 2 virtual private networks (VPNs).

Aggregated Ethernet interfaces configured with VLAN tagging can be configured with multiple logical interfaces. The only encapsulation available for aggregated Ethernet logical interfaces is **vlan-ccc**. When you configure the **vlan-id** statement, you are limited to VLAN IDs 512 through 4094.

Aggregated Ethernet interfaces configured without VLAN tagging can be configured only with the **ethernet-ccc** encapsulation. All untagged Ethernet packets received are forwarded based on the CCC parameters.

To configure aggregated Ethernet interfaces for CCC connections, include the **ae0** statement at the **[edit interfaces]** hierarchy level:

```
[edit interfaces]
ae0 {
  encapsulation (ethernet-ccc | extended-vlan-ccc | vlan-ccc);
  vlan-tagging;
  aggregated-ether-options {
    minimum-links links;
    link-speed speed;
  }
  unit logical-unit-number {
    encapsulation vlan-ccc;
    vlan-id identifier;
    family ccc;
  }
}
```

Be aware of the following limitations when configuring CCC connections over aggregated Ethernet interfaces:

- If you configured load balancing between child links, be aware that a different hash key is used to distribute packets among the child links. Standard aggregated interfaces have family inet configured. An IP version 4 (IPv4) hash key (based on the Layer 3 information) is used to distribute packets among the child links. A CCC connection over an aggregated Ethernet interface has family ccc configured instead. Instead of an IPv4 hash key, an MPLS hash key (based on the destination media access control [MAC] address) is used to distributed packets among the child links.
- The extended-vlan-ccc encapsulation is not supported on the 12-port Fast Ethernet PIC and the 48-port Fast Ethernet PIC.
- The Junos OS does not support the Link Aggregation Control Protocol (LACP) when an aggregated interface is configured as a VLAN (with vlan-ccc encapsulation). LACP can be configured only when the aggregated interface is configured with the ethernet-ccc encapsulation.

For more information about how to configure aggregated Ethernet interfaces, see the *Junos OS Network Interfaces Library for Routing Devices*.

Configuring Frame Relay Encapsulation for Layer 2 Switching Cross-Connects

For Frame Relay circuits, specify the encapsulation when configuring the DLCI. Configure each DLCI as a circuit or a regular logical interface. The DLCI for regular interfaces must be from 1 through 511. For CCC interfaces, it must be from 512 through 4094.

```
interfaces {
  type-fpc/pic/port {
    unit logical-unit-number {
      dlci dlci-identifier;
      encapsulation encapsulation-type;
      point-to-point; # Default interface type
    }
  }
}
```

You can configure these statements at the following hierarchy levels:

- [edit interfaces]
- [edit logical-systems *logical-system-name* interfaces]

Configuring PPP and Cisco HDLC Encapsulation for Layer 2 Switching Cross-Connects

For PPP and Cisco HDLC circuits, specify the encapsulation in the **encapsulation** statement. This statement configures the entire physical device. For these circuits to work, you must configure a logical interface (unit 0).

```
interfaces type-fpc/pic/port {  
  encapsulation encapsulation-type;  
  unit 0;  
}
```

You can configure these statements at the following hierarchy levels:

- [edit interfaces *type-fpc/pic/port*]
- [edit logical-systems *logical-system-name* interfaces *type-fpc/pic/port*]

Configuring the CCC Connection for Layer 2 Switching Cross-Connects

To configure Layer 2 switching cross-connects, define the connection between the two circuits by including the **interface-switch** statement. You configure this connection on the router that is acting as the switch (Router B in [Figure 3 on page 9](#)). The connection joins the interface that comes from the circuit's source to the interface that leads to the circuit's destination. When you specify the interface names, include the logical portion of the name, which corresponds to the logical unit number. The cross-connect is bidirectional, so packets received on the first interface are transmitted out the second interface, and those received on the second interface are transmitted out the first.

```
interface-switch connection-name {  
  interface interface-name.unit-number;  
  interface interface-name.unit-number;  
}
```

You can include this statement at the following hierarchy levels:

- [edit protocols connections]
- [edit logical-systems *logical-system-name* protocols connections]

Configuring MPLS for Layer 2 Switching Cross-Connects

For Layer 2 switching cross-connects to work, you must enable MPLS on the router by including at least the following statements. This minimum configuration enables MPLS on a logical interface for the switching cross-connect.

Include the **family mpls** statement:

```
family mpls;
```

You can configure this statement at the following hierarchy levels:

- [edit interfaces *interface-name* unit *logical-unit-number*]
- [edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number*]

You can then specify this logical interface in the MPLS protocol configuration:

```
mpls {
  interface interface-name; # Required to enable MPLS on the interface
}
```

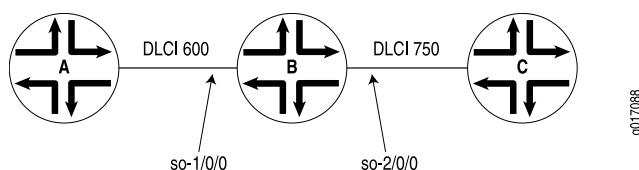
You can configure these statements at the following hierarchy levels:

- [edit protocols]
- [edit logical-systems *logical-system-name* protocols]

Example: Configuring a Layer 2 Switching Cross-Connect

Configure a full-duplex Layer 2 switching cross-connect between Router A and Router C, using a Juniper Networks router, Router B, as the virtual switch. See the topology in [Figure 4 on page 15](#) and [Figure 5 on page 16](#).

Figure 4: Topology of a Frame Relay Layer 2 Switching Cross-Connect



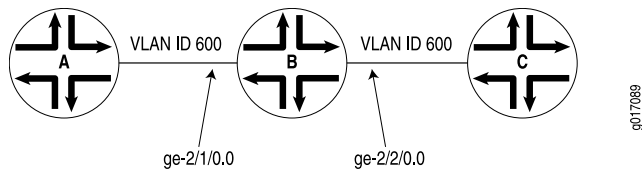
```
[edit]
interfaces {
  so-1/0/0 {
    encapsulation frame-relay-ccc;
    unit 1 {
      point-to-point;
      encapsulation frame-relay-ccc;
      dlci 600;
    }
  }
  so-2/0/0 {
    encapsulation frame-relay-ccc;
    unit 2 {
      point-to-point;
      encapsulation frame-relay-ccc;
      dlci 750;
    }
  }
}
protocols {
  connections {
    interface-switch router-a-to-router-c {
      interface so-1/0/0.1;
    }
  }
}
```

```

        interface so-2/0/0.2;
    }
}
mpls {
    interface all;
}
}

```

Figure 5: Sample Topology of a VLAN Layer 2 Switching Cross-Connect



```

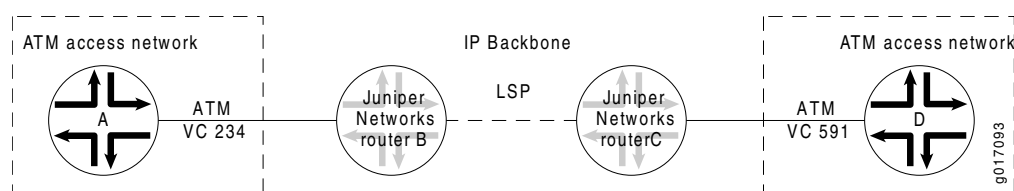
[edit]
interfaces {
    ge-2/1/0 {
        vlan-tagging;
        encapsulation vlan-ccc;
        unit 0 {
            encapsulation vlan-ccc;
            vlan-id 600;
        }
    }
    ge-2/2/0 {
        vlan-tagging;
        encapsulation vlan-ccc;
        unit 0 {
            encapsulation vlan-ccc;
            vlan-id 600;
        }
        unit 1 {
            family inet {
                vlan-id 1;
                address 10.9.200.1/24;
            }
        }
    }
}
protocols {
    mpls {
        interface all;
    }
    connections {
        interface-switch layer2-sw {
            interface ge-2/1/0.0;
            interface ge-2/2/0.0;
        }
    }
}

```

Configuring MPLS LSP Tunnel Cross-Connects Using CCC

MPLS tunnel cross-connects between interfaces and LSPs allow you to connect two distant interface circuits of the same type by creating MPLS tunnels that use LSPs as the conduit. The topology in [Figure 6 on page 17](#) illustrates an MPLS LSP tunnel cross-connect. In this topology, two separate networks, in this case ATM access networks, are connected through an IP backbone. CCC allows you to establish an LSP tunnel between the two domains. With LSP tunneling, you tunnel the ATM traffic from one network across a SONET backbone to the second network by using an MPLS LSP.

Figure 6: MPLS Tunnel Cross-Connect



When traffic from Router A (VC 234) reaches Router B, it is encapsulated and placed into an LSP, which is sent through the backbone to Router C. At Router C, the label is removed, and the packets are placed onto the ATM permanent virtual circuit (PVC) (VC 591) and sent to Router D. Similarly, traffic from Router D (VC 591) is sent over an LSP to Router B, then placed on VC 234 to Router A.

You can configure LSP tunnel cross-connect on PPP, Cisco HDLC, Frame Relay, and ATM circuits. In a single cross-connect, only like interfaces can be connected.

When you use MPLS tunnel cross-connects to support IS-IS, you must ensure that the LSP's maximum transmission unit (MTU) can, at a minimum, accommodate a 1492-octet IS-IS protocol data unit (PDU) in addition to the link-level overhead associated with the technology being connected.

For the tunnel cross-connects to work, the IS-IS frame size on the edge routers (Routers A and D in [Figure 7 on page 20](#)) must be smaller than the LSP's MTU.



NOTE: Frame size values do not include the frame check sequence (FCS) or delimiting flags.

To determine the LSP MTU required to support IS-IS, use the following calculation:

$$\text{IS-IS MTU (minimum 1492, default 1497) + frame overhead + 4 (MPLS shim header) = Minimum LSP MTU}$$

The framing overhead varies based on the encapsulation being used. The following lists the IS-IS encapsulation overhead values for various encapsulations:

- ATM
 - AAL5 multiplex—8 bytes (RFC 1483)

- VC multiplex—0 bytes
- Frame Relay
 - Multiprotocol—2 bytes (RFCs 1490 and 2427)
 - VC multiplex—0 bytes
- HDLC—4 bytes
- PPP—4 bytes
- VLAN—21 bytes (802.3/LLC)

For IS-IS to work over VLAN-CCC, the LSP's MTU must be at least 1513 bytes (or 1518 for 1497-byte PDUs). If you increase the size of a Fast Ethernet MTU above the default of 1500 bytes, you might need to explicitly configure jumbo frames on intervening equipment.

To modify the MTU, include the **mtu** statement when configuring the logical interface family at the **[edit interfaces *interface-name* unit *logical-unit-number* encapsulation *family*]** hierarchy level. For more information about setting the MTU, see the *Junos OS Network Interfaces Library for Routing Devices*.

To configure an LSP tunnel cross-connect, you must configure the following on the interdomain router (Router B in [Figure 7 on page 20](#)):

- [Configuring the CCC Encapsulation for LSP Tunnel Cross-Connects on page 18](#)
- [Configuring the CCC Connection for LSP Tunnel Cross-Connects on page 19](#)
- [Example: Configuring an LSP Tunnel Cross-Connect on page 20](#)

Configuring the CCC Encapsulation for LSP Tunnel Cross-Connects

To configure LSP tunnel cross-connects, you must configure the CCC encapsulation on the ingress and egress routers (Router B and Router C, respectively, in [Figure 7 on page 20](#)).



NOTE: You cannot configure families on CCC interfaces; that is, you cannot include the **family** statement at the **[edit interfaces *interface-name* unit *logical-unit-number*]** hierarchy level.

For PPP or Cisco HDLC circuits, include the **encapsulation** statement to configure the entire physical device. For these circuits to work, you must configure logical unit 0 on the interface.

```
type-fpc/pic/port {  
  encapsulation (ppp-ccc | cisco-hdlc-ccc);  
  unit 0;  
}
```

You can include these statements at the following hierarchy levels:

- **[edit interfaces]**
- **[edit logical-systems *logical-system-name* interfaces]**

For ATM circuits, specify the encapsulation when configuring the VC by including the following statements. For each VC, you configure whether it is a circuit or a regular logical interface.

```
at-fpc/pic/port {
  atm-options {
    vpi vpi-identifier maximum-vcs maximum-vcs;
  }
  unit logical-unit-number {
    point-to-point; # Default interface type
    encapsulation atm-ccc-vc-mux;
    vci vpi-identifier.vci-identifier;
  }
}
```

You can include these statements at the following hierarchy levels:

- [edit interfaces]
- [edit logical-systems *logical-system-name* interfaces]

For Frame Relay circuits, include the following statements to specify the encapsulation when configuring the DLCI. For each DLCI, you configure whether it is a circuit or a regular logical interface. The DLCI for regular interfaces must be in the range 1 through 511. For CCC interfaces, it must be in the range 512 through 1022.

```
type-fpc/pic/port {
  encapsulation frame-relay-ccc;
  unit logical-unit-number {
    point-to-point; # default interface type
    encapsulation frame-relay-ccc;
    dlc dlci-identifier;
  }
}
```

You can include these statements at the following hierarchy levels:

- [edit interfaces]
- [edit logical-systems *logical-system-name* interfaces]

For more information about the **encapsulation** statement, see the *Junos OS Network Interfaces Library for Routing Devices*.

Configuring the CCC Connection for LSP Tunnel Cross-Connects

To configure LSP tunnel cross-connects, include the **remote-interface-switch** statement to define the connection between the two circuits on the ingress and egress routers (Router B and Router C, respectively, in [Figure 7 on page 20](#)). The connection joins the interface or LSP that comes from the circuit's source to the interface or LSP that leads to the circuit's destination. When you specify the interface name, include the logical portion of the name, which corresponds to the logical unit number. For the cross-connect to be bidirectional, you must configure cross-connects on two routers.

```
remote-interface-switch connection-name {
  interface interface-name.unit-number;
```

```

    transmit-lsp label-switched-path;
    receive-lsp label-switched-path;
}

```

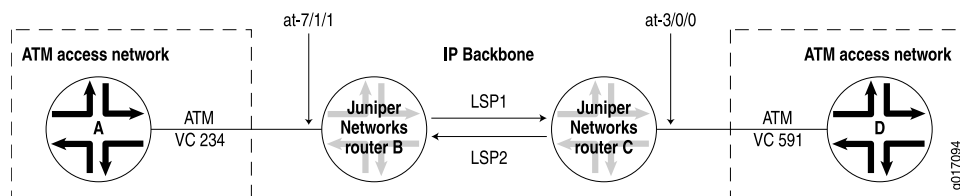
You can include these statements at the following hierarchy levels:

- [edit protocols connections]
- [edit logical-systems *logical-system-name* protocols connections]

Example: Configuring an LSP Tunnel Cross-Connect

Configure a full-duplex MPLS LSP tunnel cross-connect from Router A to Router D, passing through Router B and Router C. See the topology in [Figure 7 on page 20](#).

Figure 7: Example Topology of MPLS LSP Tunnel Cross-Connect



On Router B:

```

[edit]
interfaces {
  at-7/1/1 {
    atm-options {
      vpi 1 maximum-vcs 600;
    }
    unit 1 {
      point-to-point; # default interface type
      encapsulation atm-ccc-vc-mux;
      vci 1.234;
    }
  }
}
protocols {
  connections {
    remote-interface-switch router-b-to-router-c {
      interface at-7/1/1.1;
      transmit-lsp lsp1;
      receive-lsp lsp2;
    }
  }
}

```

On Router C:

```

[edit]
interfaces {
  at-3/0/0 {
    atm-options {
      vpi 2 maximum-vcs 600;
    }
  }
}

```

```

unit 2 {
    point-to-point; # default interface type
    encapsulation atm-ccc-vc-mux;
    vci 2.591;
}
}
protocols {
    connections {
        remote-interface-switch router-b-to-router-c {
            interface at-3/0/0.2;
            transmit-lsp lsp2;
            receive-lsp lsp1;
        }
    }
}

```

Configuring TCC

This section describes how to configure translational cross-connect (TCC). Extensive examples on how to configure TCC for interface switching and for Layer 2.5 VPNs are available in the *Junos OS, Release 14.1*.

To configure TCC, you must perform the following tasks on the router that is acting as the switch:

- [Configuring the Encapsulation for Layer 2 Switching TCCs on page 21](#)
- [Configuring the Connection for Layer 2 Switching TCCs on page 25](#)
- [Configuring MPLS for Layer 2 Switching TCCs on page 25](#)

Configuring the Encapsulation for Layer 2 Switching TCCs

To configure a Layer 2 switching TCC, specify the TCC encapsulation on the desired interfaces of the router that is acting as the switch.



NOTE: You cannot configure standard protocol families on TCC or CCC interfaces. Only the CCC family is allowed on CCC interfaces, and only the TCC family is allowed on TCC interfaces.

For Ethernet circuits and Ethernet extended VLAN circuits, you must also configure the Address Resolution Protocol (ARP). See [“Configuring ARP for Ethernet and Ethernet Extended VLAN Encapsulations” on page 24](#).

- [Configuring PPP and Cisco HDLC Encapsulation for Layer 2 Switching TCCs on page 22](#)
- [Configuring ATM Encapsulation for Layer 2 Switching TCCs on page 22](#)
- [Configuring Frame Relay Encapsulation for Layer 2 Switching TCCs on page 22](#)
- [Configuring Ethernet Encapsulation for Layer 2 Switching TCCs on page 23](#)

- [Configuring Ethernet Extended VLAN Encapsulation for Layer 2 Switching TCCs on page 23](#)
- [Configuring ARP for Ethernet and Ethernet Extended VLAN Encapsulations on page 24](#)

[Configuring PPP and Cisco HDLC Encapsulation for Layer 2 Switching TCCs](#)

For PPP and Cisco HDLC circuits, configure the encapsulation type for the entire physical device by specifying the appropriate value for the **encapsulation** statement. For these circuits to work, you must also configure the logical interface **unit 0**.

```
encapsulation (ppp-tcc | cisco-hdlc-tcc);  
unit 0{...}
```

You can include these statements at the following hierarchy levels:

- [edit interfaces *interface-name*]
- [edit logical-systems *logical-system-name* interfaces *interface-name*]

[Configuring ATM Encapsulation for Layer 2 Switching TCCs](#)

For ATM circuits, configure the encapsulation type by specifying the appropriate value for the **encapsulation** statement in the virtual circuit (VC) configuration. Specify whether each VC is a circuit or a regular logical interface.

```
atm-options {  
  vpi vpi-identifier maximum-vcs maximum-vcs;  
}  
unit logical-unit-number {  
  encapsulation (atm-tcc-vc-mux | atm-tcc-snap);  
  point-to-point;  
  vci vpi-identifier.vci-identifier;  
}
```

You can include these statements at the following hierarchy levels:

- [edit interfaces *at-fpc/pic/port*]
- [edit logical-systems *logical-system-name* interfaces *at-fpc/pic/port*]

[Configuring Frame Relay Encapsulation for Layer 2 Switching TCCs](#)

For Frame Relay circuits, configure the encapsulation type by specifying the value **frame-relay-tcc** for the **encapsulation** statement when configuring the data-link connection identifier (DLCI). You configure each DLCI as a circuit or a regular logical interface. The DLCI for regular interfaces must be in the range from 1 through 511, but for TCC and CCC interfaces it must be in the range from 512 through 1022.

```
encapsulation frame-relay-tcc;  
unit logical-unit-number {  
  dlci dlci-identifier;  
  encapsulation frame-relay-tcc;  
  point-to-point;  
}
```

You can include these statements at the following hierarchy levels:

- [edit interfaces *interface-name*]
- [edit logical-systems *logical-system-name* interfaces *interface-name*]

Configuring Ethernet Encapsulation for Layer 2 Switching TCCs

For Ethernet TCC circuits, configuring the encapsulation type for the entire physical device by specifying the value **ethernet-tcc** for the **encapsulation** statement.

You must also specify static values for a remote address and a proxy address at the [edit interfaces *interface-name* unit *unit-number* family **tcc**] or [edit logical-systems *logical-system-name* interfaces *interface-name* unit *unit-number* family **tcc**] hierarchy level.

The remote address is associated with the TCC switching router's Ethernet neighbor; in the **remote** statement you must specify both the IP address and the media access control (MAC) address of the Ethernet neighbor. The proxy address is associated with the TCC router's other neighbor connected by the unlike link; in the **proxy** statement you must specify the IP address of the non-Ethernet neighbor.

You can configure Ethernet TCC encapsulation for the interfaces on 1-port Gigabit Ethernet, 2-port Gigabit Ethernet, 4-port Fast Ethernet, and 4-port Gigabit Ethernet PICs.

```
encapsulation ethernet-tcc;
unit logical-unit-number {
  family tcc {
    proxy {
      inet-address ip-address;
    }
    remote {
      inet-address ip-address;
      mac-address mac-address;
    }
  }
}
```

You can include these statements at the following hierarchy levels:

- [edit interfaces (fe | ge)-*fpc/pic/port*]
- [edit logical-systems *logical-system-name* interfaces (fe | ge)-*fpc/pic/port*]



NOTE: For Ethernet circuits, you must also configure the Address Resolution Protocol (ARP). See “Configuring ARP for Ethernet and Ethernet Extended VLAN Encapsulations” on page 24.

Configuring Ethernet Extended VLAN Encapsulation for Layer 2 Switching TCCs

For Ethernet extended VLAN circuits, configure the encapsulation type for the entire physical device by specifying the value **extended-vlan-tcc** for the **encapsulation** statement.

You must also enable VLAN tagging. Ethernet interfaces in VLAN mode can have multiple logical interfaces. With encapsulation type **extended-vlan-tcc**, all VLAN IDs from 0 through

4094 are valid, up to a maximum of 1024 VLANs. As with Ethernet circuits, you must also specify a proxy address and a remote address at the `[edit interfaces interface-name unit logical-unit-number family tcc]` or `[edit logical-systems logical-system-name interfaces interface-name unit unit-number family tcc]` hierarchy level (see “[Configuring Ethernet Encapsulation for Layer 2 Switching TCCs](#)” on page 23).

```
encapsulation extended-vlan-tcc;
vlan-tagging;
unit logical-unit-number {
    vlan-id identifier;
    family tcc;
    proxy {
        inet-address ip-address;
    }
    remote {
        inet-address ip-address;
        mac-address mac-address;
    }
}
```

You can configure these statements at the following hierarchy levels:

- `[edit interfaces interface-name]`
- `[edit logical-systems logical-system-name interfaces interface-name]`



NOTE: For Ethernet extended VLAN circuits, you must also configure the Address Resolution Protocol (ARP). See “[Configuring ARP for Ethernet and Ethernet Extended VLAN Encapsulations](#)” on page 24.

Configuring ARP for Ethernet and Ethernet Extended VLAN Encapsulations

For Ethernet and Ethernet extended VLAN circuits with TCC encapsulation, you must also configure ARP. Because TCC simply removes one Layer 2 header and adds another, the default form of dynamic ARP is not supported; you must configure static ARP.

Because remote and proxy addresses are specified on the router performing TCC switching, you must apply the static ARP statement to the Ethernet-type interfaces of the routers that connect to the TCC-switched router. The `arp` statement must specify the IP address and the MAC address of the remotely connected neighbor by use of the unlike Layer 2 protocol on the far side of the TCC switching router.

```
arp ip-address mac mac-address;
```

You can include this statement at the following hierarchy levels:

- `[edit interfaces interface-name unit logical-unit-number family inet address ip-address]`
- `[edit logical-systems logical-system-name interfaces interface-name unit logical-unit-number family inet address ip-address]`

Configuring the Connection for Layer 2 Switching TCCs

You must configure the connection between the two circuits of the Layer 2 switching TCC on the router acting as the switch. The connection joins the interface coming from the circuit's source to the interface leading to the circuit's destination. When you specify the interface names, include the logical portion of the name, which corresponds to the logical unit number. The cross-connect is bidirectional, so packets received on the first interface are transmitted from the second interface, and those received on the second interface are transmitted from the first.

To configure a connection for a local interface switch, include the following statements:

```
interface-switch connection-name {
    interface interface-name.unit-number;
}
lsp-switch connection-name {
    transmit-lsp lsp-number;
    receive-lsp lsp-number;
}
```

You can include these statements at the following hierarchy levels:

- [edit protocols connections]
- [edit logical-systems *logical-system-name* protocols connections]

To configure a connection for a remote interface switch, include the following statements:

```
remote-interface-switch connection-name {
    interface interface-name.unit-number;
    interface interface-name.unit-number;
    transmit-lsp lsp-number;
    receive-lsp lsp-number;
}
```

You can include these statements at the following hierarchy levels:

- [edit protocols connections]
- [edit logical-systems *logical-system-name* protocols connections]

Configuring MPLS for Layer 2 Switching TCCs

For a Layer 2 switching TCC to work, you must enable MPLS on the router by including at least the following statements. This minimum configuration enables MPLS on a logical interface for the switching cross-connect.

Include the **family mpls** statement:

```
family mpls;
```

You can configure this statement at the following hierarchy levels:

- [edit interfaces *interface-name* unit *logical-unit-number*]

- [edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number*]

You can then specify this logical interface in the MPLS protocol configuration:

```
mpls {  
  interface interface-name; # Required to enable MPLS on the interface  
}
```

You can configure these statements at the following hierarchy levels:

- [edit protocols]
- [edit logical-systems *logical-system-name* protocols]



NOTE: MPLS LSP link protection does not support TCC.

Configuring CCC and TCC Graceful Restart

To enable CCC and TCC graceful restart, include the **graceful-restart** statement:

```
graceful-restart;
```

You can include this statement at the following hierarchy levels:

- [edit routing-options]
- [edit logical-systems *logical-system-name* routing-options]

The **graceful-restart** statement enables graceful restart for all protocols supporting this feature on the router. For more information about graceful restart, see the *Junos OS Routing Protocols Library for Routing Devices*.

CCC and TCC graceful restart depend on RSVP graceful restart. If you disable RSVP graceful restart, CCC and TCC graceful restart will not work. For more information about RSVP graceful restart, see *RSVP Graceful Restart* and *Configuring RSVP Graceful Restart*.

Configuring CCC Switching for Point-to-Multipoint LSPs

You can configure circuit cross-connect (CCC) between two circuits to switch traffic from interfaces to point-to-multipoint LSPs. This feature is useful for handling multicast or broadcast traffic (for example, a digital video stream).

To configure CCC switching for point-to-multipoint LSPs, you do the following:

- On the ingress provider edge (PE) router, you configure CCC to switch traffic from an incoming interface to a point-to-multipoint LSP.
- On the egress PE, you configure CCC to switch traffic from an incoming point-to-multipoint LSP to an outgoing interface.

The CCC connection for point-to-multipoint LSPs is unidirectional.

For more information about point-to-multipoint LSPs, see *Point-to-Multipoint LSPs Overview*.

To configure a CCC connection for a point-to-multipoint LSP, complete the steps in the following sections:

- [Configuring the Point-to-Multipoint LSP Switch on Ingress PE Routers on page 27](#)
- [Configuring Local Receivers on a Point-to-Multipoint CCC LSP Switch on Ingress PE Routers on page 27](#)
- [Configuring the Point-to-Multipoint LSP Switch on Egress PE Routers on page 28](#)

Configuring the Point-to-Multipoint LSP Switch on Ingress PE Routers

To configure the ingress PE router with a CCC switch for a point-to-multipoint LSP, include the **p2mp-transmit-switch** statement:

```
p2mp-transmit-switch switch-name {
  input-interface input-interface-name.unit-number;
  transmit-p2mp-lsp transmitting-lsp;
}
```

You can include the **p2mp-transmit-switch** statement at the following hierarchy levels:

- **[edit protocols connections]**
- **[edit logical-systems *logical-system-name* protocols connections]**

switch-name specifies the name of the ingress CCC switch.

input-interface *input-interface-name.unit-number* specifies the name of the ingress interface.

transmit-p2mp-lsp *transmitting-lsp* specifies the name of the transmitting point-to-multipoint LSP.

Configuring Local Receivers on a Point-to-Multipoint CCC LSP Switch on Ingress PE Routers

In addition to configuring an incoming CCC interface to a point-to-multipoint LSP on an ingress PE router, you can also configure CCC to switch traffic on an incoming CCC interface to one or more outgoing CCC interfaces by configuring output interfaces as local receivers.

To configure output interfaces, include the **output-interface** statement at the **[edit protocols connections p2mp-transmit-switch *p2mp-transmit-switch-name*]** hierarchy level.

```
[edit protocols connections]
p2mp-transmit-switch pc-ccc {
  input-interface fe-1/3/1.0;
  transmit-p2mp-lsp myp2mp;
  output-interface [fe-1/3/2.0 fe-1/3/3.0];
}
```

You can configure one or more output interfaces as local receivers on the ingress PE router using this statement.

Use the **show connections p2mp-transmit-switch** (**extensive** | **history** | **status**), **show route ccc <interface-name>** (**detail** | **extensive**), and **show route forwarding-table ccc <interface-name>** (**detail** | **extensive**) commands to view details of the local receiving interfaces on the ingress PE router.

Configuring the Point-to-Multipoint LSP Switch on Egress PE Routers

To configure the CCC switch for a point-to-multipoint LSP on the egress PE router, include the **p2mp-receive-switch** statement.

```
p2mp-receive-switch switch-name {  
    output-interface [ output-interface-name.unit-number ];  
    receive-p2mp-lsp receptive-lsp;  
}
```

You can include this statement at the following hierarchy levels:

- [edit protocols connections]
- [edit logical-systems *logical-system-name* protocols connections]

switch-name specifies the name of the egress CCC switch.

output-interface [output-interface-name.unit-number] specifies the name of one or more egress interfaces.

receive-p2mp-lsp receptive-lsp specifies the name of the receptive point-to-multipoint LSP.

CHAPTER 3

CCC and TCC Configuration Statements

- [connections \(Circuits\) on page 30](#)
- [encapsulation \(Logical Interface\) on page 31](#)
- [encapsulation \(Physical Interface\) on page 35](#)
- [interface-switch on page 40](#)
- [output-interface \(CCC\) on page 40](#)
- [lsp-switch on page 41](#)
- [p2mp-receive-switch on page 42](#)
- [p2mp-transmit-switch on page 43](#)
- [remote-interface-switch on page 44](#)

connections (Circuits)

Syntax

```
connections {
  interface-switch connection-name {
    interface interface-name.unit-number;
  }
  lsp-switch connection-name {
    transmit-lsp label-switched-path;
    receive-lsp label-switched-path;
  }
  p2mp-receive-switch {
    output-interface [ interface-name.unit-number ];
    receive-p2mp-lsp receiving-point-to-multipoint-lsp;
  }
  p2mp-transmit-switch {
    input-interface interface-name.unit-number;
    transmit-p2mp-lsp transmitting-point-to-multipoint-lsp;
  }
  remote-interface-switch connection-name {
    interface interface-name.unit-number;
    receive-lsp label-switched-path;
    transmit-lsp label-switched-path;
  }
}
```

Hierarchy Level [edit logical-systems *logical-system-name* protocols],
[edit protocols]

Release Information Statement introduced before Junos OS Release 7.4.

Description Define the connection between two circuits in a CCC connection.

Options The statements are explained separately.



NOTE: The edit logical-systems hierarchy is not available on QFabric systems.

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

Related Documentation

- [Configuring Layer 2 Switching Cross-Connects Using CCC on page 9](#)
- [Configuring MPLS LSP Tunnel Cross-Connects Using CCC on page 17](#)
- [Configuring TCC on page 21](#)
- [Configuring CCC Switching for Point-to-Multipoint LSPs on page 26](#)

encapsulation (Logical Interface)

| | |
|----------------------------|---|
| Syntax | encapsulation (atm-ccc-cell-relay atm-ccc-vc-mux atm-cisco-nlpid atm-mlppp-llc atm-nlpid atm-ppp-llc atm-ppp-vc-mux atm-snap atm-tcc-snap atm-tcc-vc-mux atm-vc-mux ether-over-atm-llc ether-vpls-over-atm-llc ether-vpls-over-fr ether-vpls-over-ppp ethernet ethernet-ccc ethernet-vpls ethernet-vpls-fr frame-relay-ccc frame-relay-ether-type frame-relay-ether-type-tcc frame-relay-ppp frame-relay-tcc gre-fragmentation multilink-frame-relay-end-to-end multilink-ppp ppp-over-ether ppp-over-ether-over-atm-llc vlan-bridge vlan-ccc vlan-vci-ccc vlan-tcc vlan-vpls); |
| Hierarchy Level | [edit interfaces <i>interface-name</i> unit <i>logical-unit-number</i>], [edit logical-systems <i>logical-system-name</i> interfaces <i>interface-name</i> unit <i>logical-unit-number</i>], [edit interfaces rlsq <i>number</i> unit <i>logical-unit-number</i>] |
| Release Information | Statement introduced before Junos OS Release 7.4. Statement introduced in Junos OS Release 12.1X48 for PTX Series Packet Transport Routers (ethernet , vlan-ccc , and vlan-tcc options only). Statement introduced in Junos OS Release 12.2 for the ACX Series Universal Access Routers. Only the atm-ccc-cell-relay and atm-ccc-vc-mux options are supported on ACX Series routers. |
| Description | Configure a logical link-layer encapsulation type. |
| Options | <p>atm-ccc-cell-relay—Use ATM cell-relay encapsulation.</p> <p>atm-ccc-vc-mux—Use ATM virtual circuit (VC) multiplex encapsulation on CCC circuits. When you use this encapsulation type, you can configure the ccc family only.</p> <p>atm-cisco-nlpid—Use Cisco ATM network layer protocol identifier (NLPID) encapsulation. When you use this encapsulation type, you can configure the inet family only.</p> <p>atm-mlppp-llc—For ATM2 IQ interfaces only, use Multilink Point-to-Point (MLPPP) over AAL5 LLC. For this encapsulation type, your router must be equipped with a Link Services or Voice Services PIC. MLPPP over ATM encapsulation is not supported on ATM2 IQ OC48 interfaces.</p> <p>atm-nlpid—Use ATM NLPID encapsulation. When you use this encapsulation type, you can configure the inet family only.</p> <p>atm-ppp-llc—(ATM2 IQ interfaces and MX Series routers with MPC/MIC interfaces using the ATM MIC with SFP only) Use PPP over AAL5 LLC encapsulation.</p> <p>atm-ppp-vc-mux—(ATM2 IQ interfaces and MX Series routers with MPC/MIC interfaces using the ATM MIC with SFP only) Use PPP over ATM AAL5 multiplex encapsulation.</p> <p>atm-snap—(All interfaces including MX Series routers with MPC/MIC interfaces using the ATM MIC with SFP) Use ATM subnetwork attachment point (SNAP) encapsulation.</p> <p>atm-tcc-snap—Use ATM SNAP encapsulation on translational cross-connect (TCC) circuits.</p> |

atm-tcc-vc-mux—Use ATM VC multiplex encapsulation on TCC circuits. When you use this encapsulation type, you can configure the **tcc** family only.

atm-vc-mux—(All interfaces including MX Series routers with MPC/MIC interfaces using the ATM MIC with SFP) Use ATM VC multiplex encapsulation. When you use this encapsulation type, you can configure the **inet** family only.

ether-over-atm-llc—(All IP interfaces including MX Series routers with MPC/MIC interfaces using the ATM MIC with SFP) For interfaces that carry IP traffic, use Ethernet over ATM LLC encapsulation. When you use this encapsulation type, you cannot configure multipoint interfaces.

ether-vpls-over-atm-llc—For ATM2 IQ interfaces only, use the Ethernet virtual private LAN service (VPLS) over ATM LLC encapsulation to bridge Ethernet interfaces and ATM interfaces over a VPLS routing instance (as described in RFC 2684, *Multiprotocol Encapsulation over ATM Adaptation Layer 5*). Packets from the ATM interfaces are converted to standard ENET2/802.3 encapsulated Ethernet frames with the frame check sequence (FCS) field removed.

ether-vpls-over-fr—For E1, T1, E3, T3, and SONET interfaces only, use the Ethernet virtual private LAN service (VPLS) over Frame Relay encapsulation to support Bridged Ethernet over Frame Relay encapsulated TDM interfaces for VPLS applications, per RFC 2427, *Multiprotocol Interconnect over Frame Relay*.



NOTE: The SONET/SDH OC3/STM1 (Multi-Rate) MIC with SFP, the Channelized SONET/SDH OC3/STM1 (Multi-Rate) MIC with SFP, and the DS3/E3 MIC do not support Ethernet over Frame Relay encapsulation.

ether-vpls-over-ppp—For E1, T1, E3, T3, and SONET interfaces only, use the Ethernet virtual private LAN service (VPLS) over Point-to-Point Protocol (PPP) encapsulation to support Bridged Ethernet over PPP-encapsulated TDM interfaces for VPLS applications.

ethernet—Use Ethernet II encapsulation (as described in RFC 894, *A Standard for the Transmission of IP Datagrams over Ethernet Networks*).

ethernet-ccc—Use Ethernet CCC encapsulation on Ethernet interfaces.

ethernet-vpls—Use Ethernet VPLS encapsulation on Ethernet interfaces that have VPLS enabled and that must accept packets carrying standard Tag Protocol ID (TPID) values.



NOTE: The built-in Gigabit Ethernet PIC on an M7i router does not support extended VLAN VPLS encapsulation.

ethernet-vpls-fr—Use in a VPLS setup when a CE device is connected to a PE device over a time-division multiplexing (TDM) link. This encapsulation type enables the PE device to terminate the outer layer 2 Frame Relay connection, use the 802.1p bits inside the inner Ethernet header to classify the packets, look at the MAC address from the Ethernet header, and use the MAC address to forward the packet into a given VPLS instance.

frame-relay-ccc—Use Frame Relay encapsulation on CCC circuits. When you use this encapsulation type, you can configure the **ccc** family only.

frame-relay-ether-type—Use Frame Relay ether type encapsulation for compatibility with Cisco Frame Relay. The physical interface must be configured with flexible-frame-relay encapsulation.

frame-relay-ether-type-tcc—Use Frame Relay ether type TCC for Cisco-compatible Frame Relay on TCC circuits to connect different media. The physical interface must be configured with flexible-frame-relay encapsulation.

frame-relay-ppp—Use PPP over Frame Relay circuits. When you use this encapsulation type, you can configure the **ppp** family only. J Series routers do not support frame-relay-ppp encapsulation.

frame-relay-tcc—Use Frame Relay encapsulation on TCC circuits for connecting different media. When you use this encapsulation type, you can configure the **tcc** family only.

gre-fragmentation—For adaptive services interfaces only, use GRE fragmentation encapsulation to enable fragmentation of IPv4 packets in GRE tunnels. This encapsulation clears the do not fragment (DF) bit in the packet header. If the packet's size exceeds the tunnel's maximum transmission unit (MTU) value, the packet is fragmented before encapsulation.

multilink-frame-relay-end-to-end—Use MLFR FRF.15 encapsulation. This encapsulation is used only on multilink, link services, and voice services interfaces and their constituent T1 or E1 interfaces, and is supported on LSQ and redundant LSQ interfaces.

multilink-ppp—Use MLPPP encapsulation. This encapsulation is used only on multilink, link services, and voice services interfaces and their constituent T1 or E1 interfaces.

ppp-over-ether—For underlying Ethernet interfaces on J Series routers, use PPP over Ethernet encapsulation. When you use this encapsulation type, you cannot configure the interface address. Instead, configure the interface address on the PPP interface. You also use PPP over Ethernet encapsulation to configure an underlying Ethernet interface for a dynamic PPPoE logical interface on M120 and M320 routers with Intelligent Queuing 2 (IQ2) PICs, and on MX Series routers with MPCs.

ppp-over-ether-over-atm-llc—(J Series routers and MX Series routers with MPCs using the ATM MIC with SFP only) For underlying ATM interfaces, use PPP over Ethernet over ATM LLC encapsulation. When you use this encapsulation type, you cannot configure the interface address. Instead, configure the interface address on the PPP interface.

vlan-bridge—Use Ethernet VLAN bridge encapsulation on Ethernet interfaces that have IEEE 802.1Q tagging, flexible-ethernet-services, and bridging enabled and that must accept packets carrying TPID 0x8100 or a user-defined TPID.

vlan-ccc—Use Ethernet virtual LAN (VLAN) encapsulation on CCC circuits. When you use this encapsulation type, you can configure the **ccc** family only.

vlan-vci-ccc—Use ATM-to-Ethernet interworking encapsulation on CCC circuits. When you use this encapsulation type, you can configure the **ccc** family only.

vlan-tcc—Use Ethernet VLAN encapsulation on TCC circuits. When you use this encapsulation type, you can configure the **tcc** family only.

vlan-vpls—Use Ethernet VLAN encapsulation on VPLS circuits.

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

Related Documentation

- [Configuring Layer 2 Switching Cross-Connects Using CCC on page 9](#)
- [Configuring the Encapsulation for Layer 2 Switching TCCs on page 21](#)
- [Configuring Interface Encapsulation on Logical Interfaces](#)
- [Configuring the CCC Encapsulation for LSP Tunnel Cross-Connects on page 18](#)
- [Circuit and Translational Cross-Connects Overview](#)
- [Identifying the Access Concentrator](#)
- [Configuring ATM Interface Encapsulation](#)
- [Configuring VLAN Encapsulation](#)
- [Configuring Extended VLAN Encapsulation](#)
- [Configuring ISDN Logical Interface Properties](#)
- [Configuring ATM-to-Ethernet Interworking](#)
- [Configuring Interface Encapsulation on PTX Series Packet Transport Routers](#)
- [Configuring CCC Encapsulation for Layer 2 VPNs](#)
- [Configuring TCC Encapsulation for Layer 2 VPNs and Layer 2 Circuits](#)
- [Configuring ATM for Subscriber Access](#)
- [CoS on ATM IMA Pseudowire Interfaces Overview](#)
- [Configuring Policing on an ATM IMA Pseudowire](#)

encapsulation (Physical Interface)

| | |
|----------------------------|--|
| Syntax | encapsulation (atm-ccc-cell-relay atm-pvc cisco-hdlc cisco-hdlc-ccc cisco-hdlc-tcc ethernet-bridge ethernet-ccc ethernet-over-atm ethernet-tcc ethernet-vpls ethernet-vpls-fr ether-vpls-over-atm-llc ethernet-vpls-ppp extended-frame-relay-ccc extended-frame-relay-ether-type-tcc extended-frame-relay-tcc extended-vlan-bridge extended-vlan-ccc extended-vlan-tcc extended-vlan-vpls flexible-ethernet-services flexible-frame-relay frame-relay frame-relay-ccc frame-relay-ether-type frame-relay-ether-type-tcc frame-relay-port-ccc frame-relay-tcc generic-services multilink-frame-relay-uni-nni ppp ppp-ccc ppp-tcc vlan-ccc vlan-vci-ccc vlan-vpls); |
| Hierarchy Level | [edit interfaces <i>interface-name</i>], [edit interfaces rlsq <i>number:number</i>] |
| Release Information | Statement introduced before Junos OS Release 7.4. Statement introduced in Junos OS Release 11.1 for EX Series switches. Statement introduced in Junos OS Release 12.1X48 for PTX Series Packet Transport Routers (flexible-ethernet-services , ethernet-ccc , and ethernet-tcc options only). |
| Description | Specify the physical link-layer encapsulation type. Not all encapsulation types are supported on the switches. See the switch CLI. |
| Default | ppp —Use serial PPP encapsulation. |
| Options | <p>atm-ccc-cell-relay—Use ATM cell-relay encapsulation.</p> <p>atm-pvc—Use ATM PVC encapsulation.</p> <p>cisco-hdlc—Use Cisco-compatible High-Level Data Link Control (HDLC) framing.</p> <p>cisco-hdlc-ccc—Use Cisco-compatible HDLC framing on CCC circuits.</p> <p>cisco-hdlc-tcc—Use Cisco-compatible HDLC framing on TCC circuits for connecting different media.</p> <p>ethernet-bridge—Use Ethernet bridge encapsulation on Ethernet interfaces that have bridging enabled and that must accept all packets.</p> <p>ethernet-ccc—Use Ethernet CCC encapsulation on Ethernet interfaces that must accept packets carrying standard Tag Protocol ID (TPID) values. For 8-port, 12-port, and 48-port Fast Ethernet PICs, CCC is not supported.</p> <p>ethernet-over-atm—For interfaces that carry IPv4 traffic, use Ethernet over ATM encapsulation. When you use this encapsulation type, you cannot configure multipoint interfaces. As defined in RFC 2684, <i>Multiprotocol Encapsulation over ATM Adaptation Layer 5</i>, this encapsulation type allows ATM interfaces to connect to devices that support only bridge protocol data units (BPDUs). Junos OS does not completely support bridging, but accepts BPDU packets as a default gateway. If you use the router as an edge device, then the router acts as a default gateway. It accepts Ethernet LLC/SNAP frames with IP or ARP in the payload, and drops the rest. For packets destined to the Ethernet LAN, a route lookup is done using the destination</p> |

IP address. If the route lookup yields a full address match, the packet is encapsulated with an LLC/SNAP and MAC header, and the packet is forwarded to the ATM interface.

ethernet-tcc—For interfaces that carry IPv4 traffic, use Ethernet TCC encapsulation on interfaces that must accept packets carrying standard TPID values. For 8-port, 12-port, and 48-port Fast Ethernet PICs, TCC is not supported.

ethernet-vpls—Use Ethernet VPLS encapsulation on Ethernet interfaces that have VPLS enabled and that must accept packets carrying standard TPID values. On M Series routers, except the M320 router, the 4-port Fast Ethernet TX PIC and the 1-port, 2-port, and 4-port, 4-slot Gigabit Ethernet PICs can use the Ethernet VPLS encapsulation type.

ethernet-vpls-fr—Use in a VPLS setup when a CE device is connected to a PE device over a time division multiplexing (TDM) link. This encapsulation type enables the PE device to terminate the outer Layer 2 Frame Relay connection, use the 802.1p bits inside the inner Ethernet header to classify the packets, look at the MAC address from the Ethernet header, and use the MAC address to forward the packet into a given VPLS instance.

ethernet-vpls-ppp—Use in a VPLS setup when a CE device is connected to a PE device over a time division multiplexing (TDM) link. This encapsulation type enables the PE device to terminate the outer Layer 2 PPP connection, use the 802.1p bits inside the inner Ethernet header to classify the packets, look at the MAC address from the Ethernet header, and use it to forward the packet into a given VPLS instance.

ether-vpls-over-atm-llc—For ATM intelligent queuing (IQ) interfaces only, use the Ethernet virtual private LAN service (VPLS) over ATM LLC encapsulation to bridge Ethernet interfaces and ATM interfaces over a VPLS routing instance (as described in RFC 2684, *Multiprotocol Encapsulation over ATM Adaptation Layer 5*). Packets from the ATM interfaces are converted to standard ENET2/802.3 encapsulated Ethernet frames with the frame check sequence (FCS) field removed.

extended-frame-relay-ccc—Use Frame Relay encapsulation on CCC circuits. This encapsulation type allows you to dedicate DLCIs 1 through 1022 to CCC.

extended-frame-relay-ether-type-tcc—Use extended Frame Relay ether type TCC for Cisco-compatible Frame Relay for DLCIs 1 through 1022. This encapsulation type is used for circuits with different media on either side of the connection.

extended-frame-relay-tcc—Use Frame Relay encapsulation on TCC circuits to connect different media. This encapsulation type allows you to dedicate DLCIs 1 through 1022 to TCC.

extended-vlan-bridge—Use extended VLAN bridge encapsulation on Ethernet interfaces that have IEEE 802.1Q VLAN tagging and bridging enabled and that must accept packets carrying TPID 0x8100 or a user-defined TPID.

extended-vlan-ccc—Use extended VLAN encapsulation on CCC circuits with Gigabit Ethernet and 4-port Fast Ethernet interfaces that must accept packets carrying 802.1Q values. For 8-port, 12-port, and 48-port Fast Ethernet PICs, extended VLAN CCC is not supported. For 4-port Gigabit Ethernet PICs, extended VLAN CCC is not supported.

extended-vlan-tcc—For interfaces that carry IPv4 traffic, use extended VLAN encapsulation on TCC circuits with Gigabit Ethernet interfaces on which you want to use 802.1Q tagging. For 4-port Gigabit Ethernet PICs, extended VLAN TCC is not supported.

extended-vlan-vpls—Use extended VLAN VPLS encapsulation on Ethernet interfaces that have VLAN 802.1Q tagging and VPLS enabled and that must accept packets carrying TPIDs 0x8100, 0x9100, and 0x9901. On M Series routers, except the M320 router, the 4-port Fast Ethernet TX PIC and the 1-port, 2-port, and 4-port, 4-slot Gigabit Ethernet PICs can use the Ethernet VPLS encapsulation type.



NOTE: The built-in Gigabit Ethernet PIC on an M7i router does not support extended VLAN VPLS encapsulation.

flexible-ethernet-services—For Gigabit Ethernet IQ interfaces and Gigabit Ethernet PICs with small form-factor pluggable transceivers (SFPs) (except the 10-port Gigabit Ethernet PIC and the built-in Gigabit Ethernet port on the M7i router), use flexible Ethernet services encapsulation when you want to configure multiple per-unit Ethernet encapsulations. Aggregated Ethernet bundles can use this encapsulation type. This encapsulation type allows you to configure any combination of route, TCC, CCC, Layer 2 virtual private networks (VPNs), and VPLS encapsulations on a single physical port. If you configure flexible Ethernet services encapsulation on the physical interface, VLAN IDs from 1 through 511 are no longer reserved for normal VLANs.

flexible-frame-relay—For IQ interfaces only, use flexible Frame Relay encapsulation when you want to configure multiple per-unit Frame Relay encapsulations. This encapsulation type allows you to configure any combination of TCC, CCC, and standard Frame Relay encapsulations on a single physical port. Also, each logical interface can have any DLCI value from 1 through 1022.

frame-relay—Use Frame Relay encapsulation.

frame-relay-ccc—Use Frame Relay encapsulation on CCC circuits.

frame-relay-ether-type—Use Frame Relay ether type encapsulation for compatibility with the Cisco Frame Relay.

frame-relay-ether-type-tcc—Use Frame Relay ether type TCC for Cisco-compatible Frame Relay on TCC circuits to connect different media.

frame-relay-port-ccc—Use Frame Relay port CCC encapsulation to transparently carry all the DLCIs between two customer edge (CE) routers without explicitly configuring each DLCI on the two provider edge (PE) routers with Frame Relay transport. When you use this encapsulation type, you can configure the **ccc** family only.

frame-relay-tcc—Use Frame Relay encapsulation on TCC circuits to connect different media.

generic-services—Use generic services encapsulation for services with a hierarchical scheduler.

multilink-frame-relay-uni-nni—Use MLFR UNI NNI encapsulation. This encapsulation is used on link services, voice services interfaces functioning as FRF.16 bundles, and their constituent T1 or E1 interfaces, and is supported on LSQ and redundant LSQ interfaces.

ppp—Use serial PPP encapsulation.

ppp-ccc—Use serial PPP encapsulation on CCC circuits. When you use this encapsulation type, you can configure the **ccc** family only.

ppp-tcc—Use serial PPP encapsulation on TCC circuits for connecting different media. When you use this encapsulation type, you can configure the **tcc** family only.

vlan-ccc—Use Ethernet VLAN encapsulation on CCC circuits.

vlan-vci-ccc—Use ATM-to-Ethernet interworking encapsulation on CCC circuits. When you use this encapsulation type, you can configure the **ccc** family only. All logical interfaces configured on the Ethernet interface must also have the encapsulation type set to **vlan-vci-ccc**.

vlan-vpls—Use VLAN VPLS encapsulation on Ethernet interfaces with VLAN tagging and VPLS enabled. Interfaces with VLAN VPLS encapsulation accept packets carrying standard TPID values only. On M Series routers, except the M320 router, the 4-port Fast Ethernet TX PIC and the 1-port, 2-port, and 4-port, 4-slot Gigabit Ethernet PICs can use the Ethernet VPLS encapsulation type.



NOTE:

- Label-switched interfaces (LSIs) do not support VLAN VPLS encapsulation. Therefore, you can only use VLAN VPLS encapsulation on a PE-router-to-CE-router interface and not a core-facing interface.
 - Starting with Junos OS release 13.3, a commit error occurs when you configure **vlan-vpls** encapsulation on a physical interface and configure **family inet** on one of the logical units. Previously, it was possible to commit this invalid configuration.
-

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

**Related
Documentation**

- *Configuring Interface Encapsulation on Physical Interfaces*
- *Configuring CCC Encapsulation for Layer 2 VPNs*
- [Configuring Layer 2 Switching Cross-Connects Using CCC on page 9](#)
- *Configuring TCC Encapsulation for Layer 2 VPNs and Layer 2 Circuits*
- *Configuring ATM Interface Encapsulation*
- *Configuring ATM-to-Ethernet Interworking*
- *Configuring VLAN Encapsulation*
- *Configuring Extended VLAN Encapsulation*
- *Configuring Encapsulation for Layer 2 Wholesale VLAN Interfaces*
- *Configuring Interfaces for Layer 2 Circuits*
- *Configuring Interface Encapsulation on PTX Series Packet Transport Routers*
- *Configuring an MPLS-Based Layer 2 VPN (CLI Procedure)*
- [Configuring MPLS LSP Tunnel Cross-Connects Using CCC on page 17](#)
- [Configuring TCC on page 21](#)
- *Configuring VPLS Interface Encapsulation*
- *Configuring Interfaces for VPLS Routing*
- *Defining the Encapsulation for Switching Cross-Connects*
- *Configuring Q-in-Q Tunneling (CLI Procedure)*

interface-switch

| | |
|---------------------------------|--|
| Syntax | <code>interface-switch <i>connection-name</i> { interface <i>interface-name.unit-number</i>; }</code> |
| Hierarchy Level | [edit logical-systems <i>logical-system-name</i> protocols connections], [edit protocols connections] |
| Release Information | Statement introduced before Junos OS Release 7.4. |
| Description | <p>Configure Layer 2 switching cross-connects. The cross-connect is bidirectional, so packets received on the first interface are transmitted out the second interface, and those received on the second interface are transmitted out the first.</p> <p>For Layer 2 switching cross-connects to work, you must also configure MPLS.</p> |
| Options | <p><i>connection-name</i>—Connection name (up to 128 characters in Junos 12.3 and later).</p> <p><i>interface interface-name.unit-number</i>—Interface name. Include the logical portion of the name, which corresponds to the logical unit number.</p> |
| Required Privilege Level | routing—To view this statement in the configuration. routing-control—To add this statement to the configuration |
| Related Documentation | <ul style="list-style-type: none">• Configuring the CCC Connection for Layer 2 Switching Cross-Connects on page 14• <i>Defining the Connection for Switching Cross-Connects</i>• <i>Junos OS MPLS Applications Library for Routing Devices</i> |

output-interface (CCC)

| | |
|---------------------------------|---|
| Syntax | <code>output-interface [<i>interface-name 1 interface-name n</i>];</code> |
| Hierarchy Level | [edit protocols connections p2mp-transmit-switch <i>p2mp-transmit-switch-name</i>] |
| Release Information | Statement introduced in Junos OS Release 12.3. |
| Description | Specify one or more output interfaces to switch traffic on an incoming CCC interface to one or more outgoing CCC interfaces. |
| Required Privilege Level | routing—To view this statement in the configuration. routing-control—To add this statement to the configuration. |
| Related Documentation | <ul style="list-style-type: none">• Configuring CCC Switching for Point-to-Multipoint LSPs on page 26 |

lsp-switch

| | |
|---------------------------------|--|
| Syntax | <code>lsp-switch <i>connection-name</i> { transmit-lsp <i>label-switched-path</i>; receive-lsp <i>label-switched-path</i>; }</code> |
| Hierarchy Level | [edit logical-systems <i>logical-system-name</i> protocols connections], [edit protocols connections] |
| Release Information | Statement introduced before Junos OS Release 7.4. |
| Description | Configure Layer 2 switching cross-connects. |
| Options | <i>connection-name</i> —Connection name. <i>receive-lsp label-switched-path</i> —Name of the LSP from the connection's source. <i>transmit-lsp label-switched-path</i> —Name of the LSP to the connection's destination. |
| Required Privilege Level | routing—To view this statement in the configuration. routing-control—To add this statement to the configuration. |
| Related Documentation | <ul style="list-style-type: none">• Configuring the Connection for Layer 2 Switching TCCs on page 25 |

p2mp-receive-switch

| | |
|---------------------------------|---|
| Syntax | <pre>p2mp-receive-switch <i>point-to-multipoint-switch-name</i> { output-interface [<i>interface-name.unit-number</i>]; receive-p2mp-lsp <i>receiving-point-to-multipoint-lsp</i>; }</pre> |
| Hierarchy Level | [edit logical-systems <i>logical-system-name</i> protocols connections], [edit protocols connections] |
| Release Information | Statement introduced before Junos OS Release 7.4. |
| Description | Configure the CCC switch for a point-to-multipoint LSP on the egress PE router. |
| Options | <p><i>point-to-multipoint-switch-name</i>—Point-to-multipoint CCC receive switch name.</p> <p><i>output-interface interface-name.unit-number</i>—Name of the egress interfaces for the point-to-multipoint LSP traffic. You can configure multiple output interfaces.</p> <p><i>receive-p2mp-lsp receiving-point-to-multipoint-lsp</i>—Name of the point-to-multipoint LSP that is switched to the output interface.</p> |
| Required Privilege Level | routing—To view this statement in the configuration. routing-control—To add this statement to the configuration. |
| Related Documentation | <ul style="list-style-type: none">• Configuring the Point-to-Multipoint LSP Switch on Egress PE Routers on page 28 |

p2mp-transmit-switch

| | |
|---------------------------------|--|
| Syntax | <code>p2mp-transmit-switch <i>point-to-multipoint-transmit-switch-name</i> { input-interface <i>interface-name.unit-number</i>; transmit-p2mp-lsp <i>transmitting-point-to-multipoint-lsp</i>; }</code> |
| Hierarchy Level | [edit logical-systems <i>logical-system-name</i> protocols connections], [edit protocols connections] |
| Release Information | Statement introduced before Junos OS Release 7.4. |
| Description | Configure the CCC switch for the point-to-multipoint LSP on the ingress PE router. |
| Options | <p><i>point-to-multipoint-transmit-switch-name</i>—Point-to-multipoint CCC transmit switch name.</p> <p><i>input-interface interface-name.unit-number</i>—Specify the name of the interface carrying incoming traffic to be switched to the point-to-multipoint LSP.</p> <p><i>transmit-p2mp-lsp transmitting-point-to-multipoint-lsp</i>—Specify the name of the point-to-multipoint LSP carrying traffic to the CCC switch on the egress PE router.</p> |
| Required Privilege Level | <p>routing—To view this statement in the configuration.</p> <p>routing-control—To add this statement to the configuration.</p> |
| Related Documentation | <ul style="list-style-type: none"> • Configuring the Point-to-Multipoint LSP Switch on Ingress PE Routers on page 27 |

remote-interface-switch

| | |
|---------------------------------|--|
| Syntax | <pre>remote-interface-switch <i>connection-name</i> { interface <i>interface-name.unit-number</i>; transmit-lsp <i>label-switched-path</i>; receive-lsp <i>label-switched-path</i>; }</pre> |
| Hierarchy Level | [edit logical-systems <i>logical-system-name</i> protocols connections], [edit protocols connections] |
| Release Information | Statement introduced before Junos OS Release 7.4. |
| Description | Configure MPLS LSP tunnel cross-connects. |
| Options | <p><i>connection-name</i>—Connection name.</p> <p><i>interface interface-name.unit-number</i>—Interface name. Include the logical portion of the name, which corresponds to the logical unit number.</p> <p><i>receive-lsp label-switched-path</i>—Name of the LSP from the connection's source.</p> <p><i>transmit-lsp label-switched-path</i>—Name of the LSP to the connection's destination.</p> |
| Required Privilege Level | routing—To view this statement in the configuration. routing-control—To add this statement to the configuration. |
| Related Documentation | <ul style="list-style-type: none">• Configuring MPLS LSP Tunnel Cross-Connects Using CCC on page 17 |

PART 3

Administration

- [CCC and TCC Operational Commands on page 47](#)

CHAPTER 4

CCC and TCC Operational Commands

- `show connections`
- `show mpls lsp`
- `show route ccc`
- `show route forwarding-table`

show connections

| | |
|------------------------------------|--|
| Syntax | <pre>show connections <brief extensive> <all interface-switch lsp-switch p2mp-receive-switch p2mp-transmit-switch remote-interface-switch> <down up up-down> <history> <labels> <logical-system (all <i>logical-system-name</i>)> <name> <status></pre> |
| Syntax (EX Series Switches) | <pre>show connections <brief extensive> <all interface-switch lsp-switch p2mp-receive-switch p2mp-transmit-switch remote-interface-switch> <down up up-down> <history> <labels> <name> <status></pre> |
| Release Information | Command introduced before Junos OS Release 7.4. Command introduced in Junos OS Release 9.5 for EX Series switches. |
| Description | Display information about the configured circuit cross-connect (CCC) connections. |
| Options | <p>none—Display the standard level of output for all configured CCC connections.</p> <p>all—(Optional) Display all connections.</p> <p>brief extensive—(Optional) Display the specified level of output. Use history to display information about connection history. Use labels to display labels used for transmit and receive LSPs. Use status to display information about the connection and interface status.</p> <p>interface-switch—(Optional) Display interface switch connections only.</p> <p>lsp-switch—(Optional) Display LSP switch connections only.</p> <p>p2mp-receive-switch—(Optional) Display point-to-multipoint LSP to local interfaces switch connections only.</p> <p>p2mp-transmit-switch—(Optional) Display local interface to point-to-multipoint LSP switch connections only.</p> <p>remote-interface-switch—(Optional) Display remote interface switch connections only.</p> <p>down up up-down—(Optional) Display nonoperational, operational, or both kinds of connections.</p> <p>history—(Optional) Display information about connection history.</p> |

labels—(Optional) Display labels used for transmit and receive.

logical-system (**all** | *logical-system-name*)—(Optional) Perform this operation on all logical systems or on a particular logical system.

name—(Optional) Display information about the specified connection only.

status—(Optional) Display information about the connection and interface status.

Required Privilege Level view

Output Fields [Table 3 on page 49](#) describes the output fields for the **show connections** command. Output fields are listed in the approximate order in which they appear.

Table 3: show connections Output Fields

| Field Name | Field Description |
|--|---|
| CCC and TCC connections [Link Monitoring On Off] | Whether link monitoring is enabled: On or Off . |
| Legend for Status (St) | Connection or circuit status. See the output's legend for an explanation of the status field values. |
| Legend for connection types | Type of connection: <ul style="list-style-type: none"> if-sw—Layer 2 switching cross-connect. rmt-if—Remote interface switch. While graceful restart is in progress, rmt-if will display a state (St) of Restart. lsp-sw—LSP stitching cross-connect. While graceful restart is in progress, lsp-sw will display a state (St) of Restart. |
| Legend for circuit types | Type of circuits: <ul style="list-style-type: none"> intf—Interface circuit. tlsp—Transmit LSP circuit. rlsp—Receive LSP circuit. |
| Connection/Circuit | Name of the configured CCC connection. |
| Type | Type of connection. |
| St | State of the connection. |
| Time last up | Time that the connection or circuit last transitioned to the Up (operational) state. |
| # Up trans | Number of times that the connection or circuit has transitioned to the Up (operational) state. |

Sample Output

show connections

```

user@switch> show connections
CCC and TCC connections [Link Monitoring On]
  Legend for status (St)
  UN -- uninitialized
  NP -- not present
  WE -- wrong encapsulation
  DS -- disabled
  Dn -- down
  -> -- only outbound conn is up
  <- -- only inbound conn is up
  Up -- operational
  RmtDn -- remote CCC down
  Restart -- restarting

  Legend for connection types
  if-sw: interface switching
  rmt-if: remote interface switching
  lsp-sw: LSP switching

  Legend for circuit types
  intf -- interface
  tlsp -- transmit LSP
  rlsp -- receive LSP

CCC Graceful restart : Restarting

Connection/Circuit      Type   St      Time last up    # Up trans
IFSW-ed                 if-sw  Up       Aug  5 15:39:15      1
  so-1/0/2.0             intf   Up
  t1-0/1/2.0             intf   Up
SW-db                   rmt-if Restart      0
  so-1/0/3.0             intf   Up
  pro4-ca                tlsp   Dn
  pro4-ac                rlsp   NP

```

show mpls lsp

| | |
|------------------------------------|---|
| Syntax | <pre>show mpls lsp <brief detail extensive terse> <autobandwidth> <bidirectional unidirectional> <bypass> <count-active-routes> <defaults> <descriptions> <down up> <externally-controlled> <externally-provisioned> <logical-system (all <i>logical-system-name</i>)> <lsp-type> <name <i>name</i>> <p2mp> <statistics> <transit></pre> |
| Syntax (EX Series Switches) | <pre>show mpls lsp <brief detail extensive terse> <bidirectional unidirectional> <bypass> <descriptions> <down up> <externally-controlled> <externally-provisioned> <lsp-type> <name <i>name</i>> <p2mp> <statistics> <transit></pre> |
| Release Information | <p>Command introduced before Junos OS Release 7.4.</p> <p>defaults option added in Junos OS Release 8.5.</p> <p>Command introduced in Junos OS Release 9.5 for EX Series switches.</p> <p>autobandwidth option added in Junos OS Release 11.4.</p> <p>externally-controlled option added in Junos OS Release 12.3.</p> <p>externally-provisioned option added in Junos OS Release 13.3.</p> <p>Command introduced in Junos OS Release 13.2X51-D15 for QFX Series.</p> |
| Description | Display information about configured and active dynamic Multiprotocol Label Switching (MPLS) label-switched paths (LSPs). |
| Options | <p>none—Display standard information about all configured and active dynamic MPLS LSPs.</p> <p>brief detail extensive terse—(Optional) Display the specified level of output. The extensive option displays the same information as the detail option, but covers the most recent 50 events.</p> |

autobandwidth—(Optional) Display automatic bandwidth information. This option is explained separately (see **show mpls lsp autobandwidth**).

bidirectional | unidirectional—(Optional) Display bidirectional or unidirectional LSP information, respectively.

bypass—(Optional) Display LSPs used for protecting other LSPs.

count-active-routes—(Optional) Display active routes for LSPs.

defaults—(Optional) Display the MPLS LSP default settings.

descriptions—(Optional) Display the MPLS label-switched path (LSP) descriptions. To view this information, you must configure the description statement at the **[edit protocol mpls lsp]** hierarchy level. Only LSPs with a description are displayed. This command is only valid for the ingress routing device, because the description is not propagated in RSVP messages.

down | up—(Optional) Display only LSPs that are inactive or active, respectively.

externally-controlled—(Optional) Display the LSPs that are under the control of an external Path Computation Element (PCE).

externally-provisioned—(Optional) Display the LSPs that are generated dynamically and provisioned by an external Path Computation Element (PCE).

logical-system (all | *logical-system-name*)—(Optional) Perform this operation on all logical systems or on a particular logical system.

lsp-type—(Optional) Display information about a particular LSP type:

- **bypass**—Sessions for bypass LSPs.
- **egress**—Sessions that terminate on this routing device.
- **ingress**—Sessions that originate from this routing device.
- **transit**—Sessions that pass through this routing device.

name *name*—(Optional) Display information about the specified LSP or group of LSPs.

p2mp—(Optional) Display information about point-to-multipoint LSPs.

statistics—(Optional) (Ingress and transit routers only) Display accounting information about LSPs. Statistics are not available for LSPs on the egress routing device, because the penultimate routing device in the LSP sets the label to 0. Also, as the packet arrives at the egress routing device, the hardware removes its MPLS header and the packet reverts to being an IPv4 packet. Therefore, it is counted as an IPv4 packet, not an MPLS packet.



NOTE: If a bypass LSP is configured for the primary static LSP, display cumulative statistics of packets traversing through the protected LSP and bypass LSP when traffic is re-optimized when the protected LSP link is restored. (Bypass LSPs are not supported on QFX Series switches.)

When used with the **bypass** option (**show mpls lsp bypass statistics**), display statistics for the traffic that flows only through the bypass LSP.

transit—(Optional) Display LSPs transiting this routing device.

Required Privilege Level view

Related Documentation

- [clear mpls lsp](#)
- [show mpls lsp autobandwidth](#)

List of Sample Output

- [show mpls lsp defaults on page 60](#)
- [show mpls lsp descriptions on page 60](#)
- [show mpls lsp detail on page 60](#)
- [show mpls lsp extensive on page 61](#)
- [show mpls lsp ingress extensive on page 62](#)
- [show mpls lsp extensive \(automatic bandwidth adjustment enabled\) on page 63](#)
- [show mpls lsp p2mp on page 64](#)
- [show mpls lsp p2mp detail on page 64](#)
- [show mpls lsp detail count-active-routes on page 65](#)
- [show mpls lsp statistics extensive on page 65](#)

Output Fields [Table 4 on page 53](#) describes the output fields for the **show mpls lsp** command. Output fields are listed in the approximate order in which they appear.

Table 4: show mpls lsp Output Fields

| Field Name | Field Description | Level of Output |
|--------------------|---|-----------------|
| Ingress LSP | Information about LSPs on the ingress routing device. Each session has one line of output. | All levels |
| Egress LSP | Information about the LSPs on the egress routing device. MPLS learns this information by querying RSVP, which holds all the transit and egress session information. Each session has one line of output. | All levels |
| Transit LSP | Number of LSPs on the transit routing devices and the state of these paths. MPLS learns this information by querying RSVP, which holds all the transit and egress session information. | All levels |
| P2MP name | Name of the point-to-multipoint LSP. Dynamically generated P2MP LSPs used for VPLS flooding use dynamically generated P2MP LSP names. The name uses the format <i>identifier:vpls:router-id:routing-instance-name</i> . The <i>identifier</i> is automatically generated by Junos OS. | All levels |

Table 4: show mpls lsp Output Fields (*continued*)

| Field Name | Field Description | Level of Output |
|-----------------------------|---|-------------------------|
| P2MP branch count | Number of destination LSPs the point-to-multipoint LSP is transmitting to. | All levels |
| P | An asterisk (*) under this heading indicates that the LSP is a primary path. | All levels |
| address | (detail and extensive) Destination (egress routing device) of the LSP. | detail extensive |
| To | Destination (egress routing device) of the session. | brief |
| From | Source (ingress routing device) of the session. | brief detail |
| State | State of the LSP handled by this RSVP session: Up , Dn (down), or Restart . | brief detail |
| Active Route | Number of active routes (prefixes) installed in the forwarding table. For ingress LSPs, the forwarding table is the primary IPv4 table (inet.0). For transit and egress RSVP sessions, the forwarding table is the primary MPLS table (mpls.0). | detail extensive |
| Rt | Number of active routes (prefixes) installed in the routing table. For ingress RSVP sessions, the routing table is the primary IPv4 table (inet.0). For transit and egress RSVP sessions, the routing table is the primary MPLS table (mpls.0). | brief |
| P | Path. An asterisk (*) underneath this column indicates that the LSP is a primary path. | brief |
| ActivePath | (Ingress LSP) Name of the active path: Primary or Secondary . | detail extensive |
| LSPname | Name of the LSP. | brief detail |
| Statistics | Displays the number of packets and the number of bytes transmitted over the LSP. These counters are reset to zero whenever the LSP path is optimized (for example, during an automatic bandwidth allocation). | extensive |
| Aggregate statistics | Displays the number of packets and the number of bytes transmitted over the LSP. These counters continue to iterate even if the LSP path is optimized. You can reset these counters to zero using the clear mpls lsp statistics command. | extensive |
| Packets | Displays the number of packets transmitted over the LSP. | brief extensive |
| Bytes | Displays the number of bytes transmitted over the LSP. | brief extensive |
| DiffServeInfo | Type of LSP: multiclass LSP (multiclass diffServ-TE LSP) or Differentiated-Services-aware traffic engineering LSP (diffServ-TE LSP). | detail |
| LSPtype | Type of LSP: static Static configured or dynamic Dynamic configured . Also indicates if the LSP is a Penultimate hop popping LSP or an Ultimate hop popping LSP. | detail extensive |
| Bypass | (Bypass LSP) Destination address (egress routing device) for the bypass LSP. | All levels |

Table 4: show mpls lsp Output Fields (*continued*)

| Field Name | Field Description | Level of Output |
|---------------------------------|--|-------------------------|
| LSPpath | Indicates whether the RSVP session is for the primary or secondary LSP path. LSPpath can be either primary or secondary and can be displayed on the ingress, egress, and transit routing devices. | detail |
| Bidir | (GMPLS) The LSP allows data to travel in both directions between GMPLS devices. | All levels |
| Bidirectional | (GMPLS) The LSP allows data to travel both ways between GMPLS devices. | All levels |
| FastReroute desired | Fast reroute has been requested by the ingress routing device. | detail |
| Link protection desired | Link protection has been requested by the ingress routing device. | detail |
| LoadBalance | (Ingress LSP) CSPF load-balancing rule that was configured to select the LSP's path among equal-cost paths: Most-fill , Least-fill , or Random . | detail extensive |
| Signal type | Signal type for GMPLS LSPs. The signal type determines the peak data rate for the LSP: DS0 , DS3 , STS-1 , STM-1 , or STM-4 . | All levels |
| Encoding type | LSP encoding type: Packet , Ethernet , PDH , SDH/SONET , Lambda , or Fiber . | All levels |
| Switching type | Type of switching on the links needed for the LSP: Fiber , Lambda , Packet , TDM , or PSC-1 . | All levels |
| GPID | Generalized Payload Identifier (identifier of the payload carried by an LSP): HDLC , Ethernet , IPv4 , PPP , or Unknown . | All levels |
| Protection | Configured protection capability desired for the LSP: Extra , Enhanced , none , One plus one , One to one , or Shared . | All levels |
| Upstream label in | (Bidirectional LSPs) Incoming label for reverse direction traffic for this LSP. | All levels |
| Upstream label out | (Bidirectional LSPs) Outgoing label for reverse direction traffic for this LSP. | All levels |
| Suggested label received | (Bidirectional LSPs) Label the upstream node suggests to use in the Resv message that is sent. | All levels |
| Suggested label sent | (Bidirectional LSPs) Label the downstream node suggests to use in the Resv message that is returned. | All levels |
| Autobandwidth | (Ingress LSP) The LSP is performing autobandwidth allocation. | detail extensive |
| MinBW | (Ingress LSP) Configured minimum value of the LSP, in bps. | detail extensive |
| MaxBW | (Ingress LSP) Configured maximum value of the LSP, in bps. | detail extensive |

Table 4: show mpls lsp Output Fields (*continued*)

| Field Name | Field Description | Level of Output |
|---|--|-------------------------|
| Dynamic MinBW | (Ingress LSP) Displays the current dynamically specified minimum bandwidth allocation for the LSP, in bps. | detail extensive |
| Adjustment Timer | (Ingress LSP) Configured value for the adjust-timer statement, indicating the total amount of time allowed before bandwidth adjustment will take place, in seconds. | detail extensive |
| Adjustment Threshold | (Ingress LSP) Configured value for the adjust-threshold statement. Specifies how sensitive the automatic bandwidth adjustment for an LSP is to changes in bandwidth utilization. | detail extensive |
| Time for Next Adjustment | (Ingress LSP) Time in seconds until the next automatic bandwidth adjustment sample is taken. | detail extensive |
| Time of Last Adjustment | (Ingress LSP) Date and time since the last automatic bandwidth adjustment was completed. | detail extensive |
| Max AvgBW util | (Ingress LSP) Current value of the actual maximum average bandwidth utilization, in bps. | detail extensive |
| Overflow limit | (Ingress LSP) Configured value of the threshold overflow limit. | detail extensive |
| Overflow sample count | (Ingress LSP) Current value for the overflow sample count. | detail extensive |
| Bandwidth Adjustment in <i>nnn</i> second(s) | (Ingress LSP) Current value of the bandwidth adjustment timer, indicating the amount of time remaining until the bandwidth adjustment will take place, in seconds. | detail extensive |
| Underflow limit | (Ingress LSP) Configured value of the threshold underflow limit. | detail extensive |
| Underflow sample count | (Ingress LSP) Current value for the underflow sample count. | detail extensive |
| Underflow Max AvgBW | (Ingress LSP) The highest sample bandwidth among the underflow samples recorded currently. This is the signaling bandwidth if an adjustment occurs because of an underflow. | detail extensive |
| Active path indicator | (Ingress LSP) A value of * indicates that the path is active. The absence of * indicates that the path is not active. In the following example, "long" is the active path. *Primary long Standby short | detail extensive |
| Primary | (Ingress LSP) Name of the primary path. | detail extensive |
| Secondary | (Ingress LSP) Name of the secondary path. | detail extensive |
| Standby | (Ingress LSP) Name of the path in standby mode. | detail extensive |

Table 4: show mpls lsp Output Fields (*continued*)

| Field Name | Field Description | Level of Output |
|---|---|-------------------------|
| State | (Ingress LSP) State of the path: Up or Dn (down). | detail extensive |
| COS | (Ingress LSP) Class-of-service value. | detail extensive |
| Bandwidth per class | (Ingress LSP) Active bandwidth for the LSP path for each MPLS class type, in bps. | detail extensive |
| Priorities | (Ingress LSP) Configured value of the setup priority and the hold priority respectively (the setup priority is displayed first), where 0 is the highest priority and 7 is the lowest priority. If you have not explicitly configured these values, the default values are displayed (7 for the setup priority and 0 for the hold priority). | detail extensive |
| OptimizeTimer | (Ingress LSP) Configured value of the optimize timer, indicating the total amount of time allowed before path reoptimization, in seconds. | detail extensive |
| SmartOptimizeTimer | (Ingress LSP) Configured value of the smart optimize timer, indicating the total amount of time allowed before path reoptimization, in seconds. | detail extensive |
| Reoptimization in xxx seconds | (Ingress LSP) Current value of the optimize timer, indicating the amount of time remaining until the path will be reoptimized, in seconds. | detail extensive |
| Computed ERO (S [L] denotes strict [loose] hops) | (Ingress LSP) Computed explicit route. A series of hops, each with an address followed by a hop indicator. The value of the hop indicator can be strict (S) or loose (L). | detail extensive |
| CSPF metric | (Ingress LSP) Constrained Shortest Path First metric for this path. | detail extensive |

Table 4: show mpls lsp Output Fields (*continued*)

| Field Name | Field Description | Level of Output |
|---------------------|--|-------------------------------|
| Received RRO | <p>(Ingress LSP) Received record route. A series of hops, each with an address followed by a flag. (In most cases, the received record route is the same as the computed explicit route. If Received RRO is different from Computed ERO, there is a topology change in the network, and the route is taking a detour.) The following flags identify the protection capability and status of the downstream node:</p> <ul style="list-style-type: none"> • 0x01—Local protection available. The link downstream from this node is protected by a local repair mechanism. This flag can be set only if the Local protection flag was set in the SESSION_ATTRIBUTE object of the corresponding Path message. • 0x02—Local protection in use. A local repair mechanism is in use to maintain this tunnel (usually because of an outage of the link it was routed over previously). • 0x03—Combination of 0x01 and 0x02. • 0x04—Bandwidth protection. The downstream routing device has a backup path providing the same bandwidth guarantee as the protected LSP for the protected section. • 0x08—Node protection. The downstream routing device has a backup path providing protection against link and node failure on the corresponding path section. If the downstream routing device can set up only a link-protection backup path, the Local protection available bit is set but the Node protection bit is cleared. • 0x09—Detour is established. Combination of 0x01 and 0x08. • 0x10—Preemption pending. The preempting node sets this flag if a pending preemption is in progress for the traffic engine LSP. This flag indicates to the ingress legacy edge router (LER) of this LSP that it should be rerouted. • 0x20—Node ID. Indicates that the address specified in the RRO's IPv4 or IPv6 sub-object is a node ID address, which refers to the router address or router ID. Nodes must use the same address consistently. • 0xb—Detour is in use. Combination of 0x01, 0x02, and 0x08. | detail extensive |
| Index number | (Ingress LSP) Log entry number of each LSP path event. The numbers are in chronological descending order, with a maximum of 50 index numbers displayed. | extensive |
| Date | (Ingress LSP) Date of the LSP event. | extensive |
| Time | (Ingress LSP) Time of the LSP event. | extensive |
| Event | (Ingress LSP) Description of the LSP event. | extensive |
| Created | (Ingress LSP) Date and time the LSP was created. | extensive |
| Resv style | (Bypass) RSVP reservation style. This field consists of two parts. The first is the number of active reservations. The second is the reservation style, which can be FF (fixed filter), SE (shared explicit), or WF (wildcard filter). | brief detail extensive |
| Labelin | Incoming label for this LSP. | brief detail |
| Labelout | Outgoing label for this LSP. | brief detail |

Table 4: show mpls lsp Output Fields (*continued*)

| Field Name | Field Description | Level of Output |
|--------------------------------|---|---------------------|
| LSPname | Name of the LSP. | brief detail |
| Time left | Number of seconds remaining in the lifetime of the reservation. | detail |
| Since | Date and time when the RSVP session was initiated. | detail |
| Tspec | Sender's traffic specification, which describes the sender's traffic parameters. | detail |
| Port number | Protocol ID and sender or receiver port used in this RSVP session. | detail |
| PATH rcvfrom | Address of the previous-hop (upstream) routing device or client, interface the neighbor used to reach this router, and number of packets received from the upstream neighbor. | detail |
| PATH sentto | Address of the next-hop (downstream) routing device or client, interface used to reach this neighbor, and number of packets sent to the downstream routing device. | detail |
| RESV rcvfrom | Address of the previous-hop (upstream) routing device or client, interface the neighbor used to reach this routing device, and number of packets received from the upstream neighbor. The output in this field, which is consistent with that in the PATH rcvfrom field, indicates that the RSVP negotiation is complete. | detail |
| Record route | Recorded route for the session, taken from the record route object. | detail |
| Soft preempt | Number of soft preemptions that occurred on a path and when the last soft preemption occurred. Only successful soft preemptions are counted (those that actually resulted in a new path being used). | detail |
| Soft preemption pending | Path is in the process of being soft preempted. This display is removed once the ingress router has calculated a new path. | detail |
| MPLS-TE LSP Defaults | Default settings for MPLS traffic engineered LSPs: <ul style="list-style-type: none"> • LSP Holding Priority—Determines the degree to which an LSP holds on to its session reservation after the LSP has been set up successfully. • LSP Setup Priority—Determines whether a new LSP that preempts an existing LSP can be established. • Hop Limit—Specifies the maximum number of routers the LSP can traverse (including the ingress and egress). • Bandwidth—Specifies the bandwidth in bits per second for the LSP. • LSP Retry Timer—Length of time in seconds that the ingress router waits between attempts to establish the primary path. | defaults |

The XML tag name of the **bandwidth** tag under the **auto-bandwidth** tag has been updated to **maximum-average-bandwidth**. You can see the new tag when you issue the **show mpls lsp extensive** command with the **| display xml** pipe option. If you have any scripts that use the **bandwidth** tag, ensure that they are updated to **maximum-average-bandwidth**.

Sample Output

show mpls lsp defaults

```
user@host> show mpls lsp defaults
MPLS-TE LSP Defaults
  LSP Holding Priority      0
  LSP Setup Priority       7
  Hop Limit                255
  Bandwidth                0
  LSP Retry Timer          30 seconds
```

show mpls lsp descriptions

```
user@host> show mpls lsp descriptions
Ingress LSP: 3 sessions
To          LSP name          Description
10.0.0.195  to-sanjose                to-sanjose-desc
10.0.0.195  to-sanjose-other-desc      other-desc
Total 2 displayed, Up 2, Down 0
```

show mpls lsp detail

```
user@host> show mpls lsp detail
Ingress LSP: 1 sessions

192.168.0.4
  From: 192.168.0.5, State: Up, ActiveRoute: 0, LSPname: E-D
  ActivePath: (primary)
  LSPtype: Static Configured, Penultimate hop popping
  LoadBalance: Random
  Encoding type: Packet, Switching type: Packet, GPID: IPv4
  *Primary                               State: Up
    Priorities: 7 0
    SmartOptimizeTimer: 180
    Computed ERO (S [L] denotes strict [loose] hops): (CSPF metric: 30)
  10.0.0.18 S 10.0.0.22 S
    Received RRO (ProtectionFlag 1=Available 2=InUse 4=B/W 8=Node 10=SoftPreempt
  20=Node-ID):
      10.0.0.18 10.0.0.22
Total 1 displayed, Up 1, Down 0

Egress LSP: 1 sessions

192.168.0.5
  From: 192.168.0.4, LSPstate: Up, ActiveRoute: 0
  LSPname: E-D, LSPpath: Primary
  Suggested label received: -, Suggested label sent: -
  Recovery label received: -, Recovery label sent: -
  Resv style: 1 FF, Label in: 3, Label out: -
  Time left: 157, Since: Wed Jul 18 17:55:12 2012
  Tspec: rate 0bps size 0bps peak Infbps m 20 M 1500
  Port number: sender 1 receiver 46128 protocol 0
  PATH rcvfrom: 10.0.0.18 (lt-1/2/0.17) 3 pkts
  Adspec: received MTU 1500
  PATH sentto: localclient
  RESV rcvfrom: localclient
  Record route: 10.0.0.22 10.0.0.18 <self>
Total 1 displayed, Up 1, Down 0
```

Transit LSP: 0 sessions
Total 0 displayed, Up 0, Down 0

show mpls lsp extensive

user@host> show mpls lsp extensive
Ingress LSP: 4 sessions

```
1.1.1.1
  From: 3.3.3.3, State: Up, ActiveRoute: 0, LSPname: m120b-to-mx960
  ActivePath: DEFAULT (primary)
  FastReroute desired
  LSptype: Static Configured, Penultimate hop popping
  LoadBalance: Random
  Encoding type: Packet, Switching type: Packet, GPID: IPv4
  *Primary  DEFAULT      State: Up
    Priorities: 7 0
    SmartOptimizeTimer: 180
    Computed ERO (S [L] denotes strict [loose] hops): (CSPF metric: 310)
10.0.35.5 S 10.0.15.1 S
  Received RRO (ProtectionFlag 1=Available 2=InUse 4=B/W 8=Node 10=SoftPreempt
20=Node-ID):
    10.0.34.4(flag=1) 10.0.14.1
50 Sep 13 16:08:19.712 Record Route: 10.0.35.5(flag=1) 10.0.15.1
49 Sep 13 16:08:16.720 Record Route: 10.0.34.4(flag=1) 10.0.14.1
48 Sep 13 16:08:16.699 Fast-reroute Detour Up
47 Sep 13 16:08:13.702 Record Route: 10.0.34.4 10.0.14.1
46 Sep 13 16:08:13.702 Up
45 Sep 13 16:08:13.672 Originate make-before-break call
44 Sep 13 16:08:13.672 CSPF: computation result accepted 10.0.34.4 10.0.14.1

43 Sep 13 16:08:13.672 Selected as active path
42 Sep 13 16:08:13.672 Make-before-break: Switched to new instance
41 Sep 13 16:08:01.685 Pending path switchover, skip CSPF run[3 times]
40 Sep 13 16:06:33.910 Deselected as active
39 Sep 13 16:06:33.910 Pending path switchover, skip CSPF run

38 Sep 13 16:06:19.521 Record Route: 10.0.35.5 10.0.15.1
37 Sep 13 16:06:19.518 ResvTear received
36 Sep 13 16:06:19.518 Fast-reroute Detour Down
35 Sep 13 16:06:16.676 Record Route: 10.0.35.5(flag=1) 10.0.15.1
34 Sep 13 16:06:13.670 Record Route: 10.0.35.5 10.0.15.1
33 Sep 13 16:06:13.670 Up
32 Sep 13 16:06:13.569 Pending path switchover, skip CSPF run

31 Sep 13 16:06:13.569 CSPF: link down/deleted:
10.0.34.3(3.3.3.3:79)(m120-b-re1.00/3.3.3.3)->0.0.0.0(0.0.0.0:0)(m120-b-re1.04/0.0.0.0)

30 Sep 13 16:06:13.552 Pending path switchover, skip CSPF run

29 Sep 13 16:06:13.552 CSPF: link down/deleted:
0.0.0.0(0.0.0.0:0)(m120-b-re1.04/0.0.0.0)->0.0.0.0(4.4.4.4:0)(m10i-a-re0.00/4.4.4.4)

28 Sep 13 16:06:13.549 Originate make-before-break call
27 Sep 13 16:06:13.549 CSPF: computation result accepted 10.0.35.5 10.0.15.1

26 Sep 13 16:06:13.548 Tunnel local repaired
25 Sep 13 16:06:13.546 Record Route: 10.0.23.2 10.0.12.1
24 Sep 13 16:06:13.546 10.0.34.3: Tunnel local repaired
23 Sep 13 16:06:13.546 10.0.34.3: Down
22 Sep 13 16:03:46.842 Fast-reroute Detour Up
```

```

21 Sep 13 16:03:42.730 Record Route: 10.0.34.4(flag=1) 10.0.14.1
20 Sep 13 16:03:39.836 Selected as active path
19 Sep 13 16:03:39.834 Record Route: 10.0.34.4 10.0.14.1
18 Sep 13 16:03:39.834 Up
17 Sep 13 16:03:39.698 Originate Call
16 Sep 13 16:03:39.698 CSPF: computation result accepted 10.0.34.4 10.0.14.1

15 Sep 13 16:03:39.697 Clear Call
14 Sep 13 16:03:39.696 Deselected as active
13 Sep 13 16:03:37.837 Record Route: 10.0.34.4 10.0.14.1
12 Sep 13 16:03:32.829 Fast-reroute Detour Down
11 Sep 13 16:02:15.493 Record Route: 10.0.34.4(flag=1) 10.0.14.1
10 Sep 13 16:02:15.486 Fast-reroute Detour Up
9 Sep 13 16:02:12.468 Record Route: 10.0.34.4 10.0.14.1
8 Sep 13 16:02:07.460 Fast-reroute Detour Down
7 Sep 13 15:57:46.741 Fast-reroute Detour Up
6 Sep 13 15:57:40.768 Record Route: 10.0.34.4(flag=1) 10.0.14.1
5 Sep 13 15:57:37.761 Selected as active path
4 Sep 13 15:57:37.760 Record Route: 10.0.34.4 10.0.14.1
3 Sep 13 15:57:37.760 Up
2 Sep 13 15:57:37.733 Originate Call
1 Sep 13 15:57:37.733 CSPF: computation result accepted 10.0.34.4 10.0.14.1

```

Created: Fri Sep 13 15:57:38 2013

Total 1 displayed, Up 1, Down 0

Egress LSP: 4 sessions, 6 detours

Total 0 displayed, Up 0, Down 0

Transit LSP: 6 sessions, 1 detours

1.1.1.1

```

From: 3.3.3.3, LSPstate: Up, ActiveRoute: 0
LSPname: m120b-to-mx960, LSPpath: Primary
Suggested label received: -, Suggested label sent: -
Recovery label received: -, Recovery label sent: 302288
Resv style: 1 FF, Label in: 300416, Label out: 302288
Time left: 147, Since: Fri Sep 13 16:08:16 2013
Tspec: rate 0bps size 0bps peak Infbps m 20 M 1500
Port number: sender 4 receiver 13955 protocol 0
Detour branch from 10.0.34.4, to skip 1.1.1.1, Up
Tspec: rate 0bps size 0bps peak Infbps m 20 M 1500
Adspec: received MTU 1500
Path MTU: received 0
PATH rcvfrom: 10.0.34.4 (ge-4/3/7.0) 7 pkts
Adspec: received MTU 1500 sent MTU 1500
PATH sentto: 10.0.35.5 (ge-3/1/0.0) 7 pkts
RESV rcvfrom: 10.0.35.5 (ge-3/1/0.0) 7 pkts
Explicit route: 10.0.35.5 10.0.15.1
Record route: 10.0.34.3 10.0.34.4 <self>10.0.35.5 10.0.15.1
Label in: 300416, Label out: 302288
Total 1 displayed, Up 1, Down 0

```

show mpls lsp ingress extensive

```
user@host> show mpls lsp ingress extensive
```

Ingress LSP: 1 sessions

50.0.0.1

```

From: 10.0.0.1, State: Up, ActiveRoute: 0, LSPname: test
ActivePath: (primary)

```

```

LSPTYPE: Static Configured
LoadBalance: Random
Encoding type: Packet, Switching type: Packet, GPID: IPv4
*Primary                               State: Up
  Priorities: 7 0
  OptimizeTimer: 300
  SmartOptimizeTimer: 180
  Reoptimization in 240 second(s).
  Computed ERO (S [L] denotes strict [loose] hops): (CSPF metric: 3)
1.1.1.2 S 4.4.4.1 S 5.5.5.2 S
  Received RRO (ProtectionFlag 1=Available 2=InUse 4=B/W 8=Node 10=SoftPreempt
20=Node-ID):
    1.1.1.2 4.4.4.1 5.5.5.2
17 Aug 3 13:17:33.601 CSPF: computation result ignored, new path less avail
bw[3 times]
16 Aug 3 13:02:51.283 CSPF: computation result ignored, new path no benefit[2
times]
15 Aug 3 12:54:36.678 Selected as active path
14 Aug 3 12:54:36.676 Record Route: 1.1.1.2 4.4.4.1 5.5.5.2
13 Aug 3 12:54:36.676 Up
12 Aug 3 12:54:33.924 Deselected as active
11 Aug 3 12:54:33.924 Originate Call
10 Aug 3 12:54:33.923 Clear Call
9 Aug 3 12:54:33.923 CSPF: computation result accepted 1.1.1.2 4.4.4.1
5.5.5.2
8 Aug 3 12:54:33.922 2.2.2.2: No Route toward dest
7 Aug 3 12:54:28.177 CSPF: computation result ignored, new path no benefit[4
times]
6 Aug 3 12:35:03.830 Selected as active path
5 Aug 3 12:35:03.828 Record Route: 2.2.2.2 3.3.3.2
4 Aug 3 12:35:03.827 Up
3 Aug 3 12:35:03.814 Originate Call
2 Aug 3 12:35:03.814 CSPF: computation result accepted 2.2.2.2 3.3.3.2
1 Aug 3 12:34:34.921 CSPF failed: no route toward 50.0.0.1
Created: Tue Aug 3 12:34:35 2010
Total 1 displayed, Up 1, Down 0

```

show mpls lsp extensive (automatic bandwidth adjustment enabled)

```

user@host> show mpls lsp extensive
Ingress LSP: 1 sessions

192.168.0.4
  From: 192.168.0.5, State: Up, ActiveRoute: 0, LSPName: E-D
  ActivePath: (primary)
  Node/Link protection desired
  LSPTYPE: Static Configured, Penultimate hop popping
  LoadBalance: Random
  Autobandwidth
  MinBW: 300bps, MaxBW: 1000bps, Dynamic MinBW: 1000bps
  Adjustment Timer: 300 secs AdjustThreshold: 25%
  Max AvgBW util: 963.739bps, Bandwidth Adjustment in 0 second(s).
  Min BW Adjust Interval: 1000, MinBW Adjust Threshold (in %): 50
  Overflow limit: 0, Overflow sample count: 0
  Underflow limit: 0, Underflow sample count: 9, Underflow Max AvgBW: 614.421bps

  Encoding type: Packet, Switching type: Packet, GPID: IPv4
  *Primary                               State: Up
    Priorities: 7 0
    Bandwidth: 1000bps
    SmartOptimizeTimer: 180

```

```

    Computed ERO (S [L] denotes strict [loose] hops): (CSPF metric: 30)
10.0.0.18 S 10.0.0.22 S
    Received RRO (ProtectionFlag 1=Available 2=InUse 4=B/W 8=Node 10=SoftPreempt
20=Node-ID):
        192.168.0.6(flag=0x20) 10.0.0.18(Label=299792) 192.168.0.4(flag=0x20)
10.0.0.22(Label=3)
    12 Apr 30 10:25:17.024 Make-before-break: Switched to new instance
    11 Apr 30 10:25:16.023 Record Route: 192.168.0.6(flag=0x20)
10.0.0.18(Label=299792) 192.168.0.4(flag=0x20) 10.0.0.22(Label=3)
    10 Apr 30 10:25:16.023 Up
    9 Apr 30 10:25:16.023 Automatic Autobw adjustment succeeded: BW changes from
300 bps to 1000 bps
    8 Apr 30 10:25:15.946 Originate make-before-break call
    7 Apr 30 10:25:15.946 CSPF: computation result accepted 10.0.0.18 10.0.0.22

    6 Apr 30 10:16:42.891 Selected as active path
    5 Apr 30 10:16:42.891 Record Route: 192.168.0.6(flag=0x20)
10.0.0.18(Label=299776) 192.168.0.4(flag=0x20) 10.0.0.22(Label=3)
    4 Apr 30 10:16:42.890 Up
    3 Apr 30 10:16:42.828 Originate Call
    2 Apr 30 10:16:42.828 CSPF: computation result accepted 10.0.0.18 10.0.0.22

    1 Apr 30 10:16:14.064 CSPF: could not determine self[2 times]
Created: Tue Apr 30 10:15:16 2013
Total 1 displayed, Up 1, Down 0

Egress LSP: 0 sessions
Total 0 displayed, Up 0, Down 0

Transit LSP: 0 sessions
Total 0 displayed, Up 0, Down 0

```

show mpls lsp p2mp

```

user@host> show mpls lsp p2mp
Ingress LSP: 2 sessions
P2MP name: p2mp-lsp1, P2MP branch count: 1
To          From          State Rt P ActivePath      LSPname
10.255.245.51 10.255.245.50 Up    0 * path1         p2mp-branch-1
P2MP name: p2mp-lsp2, P2MP branch count: 1
To          From          State Rt P ActivePath      LSPname
10.255.245.51 10.255.245.50 Up    0 * path1         p2mp-st-br1
Total 2 displayed, Up 2, Down 0

Egress LSP: 0 sessions
Total 0 displayed, Up 0, Down 0

Transit LSP: 0 sessions
Total 0 displayed, Up 0, Down 0

```

show mpls lsp p2mp detail

```

user@host> show mpls lsp p2mp detail
Ingress LSP: 2 sessions
P2MP name: p2mp-lsp1, P2MP branch count: 1

10.255.245.51
  From: 10.255.245.50, State: Up, ActiveRoute: 0, LSPname: p2mp-branch-1
  ActivePath: path1 (primary)
  P2MP name: p2mp-lsp1
  LoadBalance: Random

```

```

    Encoding type: Packet, Switching type: Packet, GPID: IPv4
    *Primary path1 State: Up
    Computed ERO (S [L] denotes strict [loose] hops): (CSPF metric: 25)
    192.168.208.17 S
    Received RRO (ProtectionFlag 1=Available 2=InUse 4=B/W 8=Node 10=SoftPreempt):
        192.168.208.17
    P2MP name: p2mp-lsp2, P2MP branch count: 1

10.255.245.51
    From: 10.255.245.50, State: Up, ActiveRoute: 0, LSPname: p2mp-st-br1
    ActivePath: path1 (primary)
    P2MP name: p2mp-lsp2
    LoadBalance: Random
    Encoding type: Packet, Switching type: Packet, GPID: IPv4
    *Primary path1 State: Up
    Computed ERO (S [L] denotes strict [loose] hops): (CSPF metric: 25)
    192.168.208.17 S
    Received RRO (ProtectionFlag 1=Available 2=InUse 4=B/W 8=Node 10=SoftPreempt):
        192.168.208.17
Total 2 displayed, Up 2, Down 0

```

show mpls lsp detail count-active-routes

```

user@host> show mpls lsp detail count-active-routes
Ingress LSP: 1 sessions

213.119.192.2
    From: 156.154.162.128, State: Up, ActiveRoute: 1, LSPname: to-lahore
    ActivePath: (primary)
    LSPtype: Static Configured
    LoadBalance: Random
    Autobandwidth
    MinBW: 5Mbps MaxBW: 250Mbps
    Adjustment Timer: 300 secs
    Max AvgBW util: 60.2599Mbps, Bandwidth Adjustment in 0 second(s).
    Overflow limit: 0, Overflow sample count: 0
    Encoding type: Packet, Switching type: Packet, GPID: IPv4
    *Primary State: Up
        Priorities: 7 0
        Bandwidth: 5Mbps
        SmartOptimizeTimer: 180
        Computed ERO (S [L] denotes strict [loose] hops): (CSPF metric: 4)
    10.252.0.177 S
    Received RRO (ProtectionFlag 1=Available 2=InUse 4=B/W 8=Node 10=SoftPreempt
    20=Node-ID):
        10.252.0.177
Total 1 displayed, Up 1, Down 0

Egress LSP: 0 sessions
Total 0 displayed, Up 0, Down 0

Transit LSP: 0 sessions
Total 0 displayed, Up 0, Down 0

```

show mpls lsp statistics extensive

```

user@host> show mpls lsp statistics extensive
Ingress LSP: 1 sessions

```

```
192.168.0.4
  From: 192.168.0.5, State: Up, ActiveRoute: 0, LSPName: E-D
  Statistics: Packets 302, Bytes 28992
  Aggregate statistics: Packets 302, Bytes 28992
  ActivePath: (primary)
  LSPType: Static Configured, Penultimate hop popping
  LoadBalance: Random
  Encoding type: Packet, Switching type: Packet, GPID: IPv4
  *Primary                               State: Up
    Priorities: 7 0
    SmartOptimizeTimer: 180
    Computed ERO (S [L] denotes strict [loose] hops): (CSPF metric: 30)
10.0.0.18 S 10.0.0.22 S
  Received RRO (ProtectionFlag 1=Available 2=InUse 4=B/W 8=Node 10=SoftPreempt
20=Node-ID):
    10.0.0.18 10.0.0.22
  6 Oct  3 11:18:28.281 Selected as active path
  5 Oct  3 11:18:28.281 Record Route:  10.0.0.18 10.0.0.22
  4 Oct  3 11:18:28.280 Up
  3 Oct  3 11:18:27.995 Originate Call
  2 Oct  3 11:18:27.995 CSPF: computation result accepted  10.0.0.18 10.0.0.22

  1 Oct  3 11:17:59.118 CSPF failed: no route toward 192.168.0.4[2 times]
Created: Wed Oct  3 11:17:01 2012
Total 1 displayed, Up 1, Down 0
```

show route ccc

| | |
|---------------------------------|---|
| Syntax | show route ccc ccc <brief detail extensive terse> <logical-system (all <i>logical-system-name</i>)> |
| Release Information | Command introduced before Junos OS Release 7.4. |
| Description | Display circuit cross-connect (CCC) entries in the Multiprotocol Link Switching (MPLS) routing table. |
| Options | <p>ccc—Name of an entry with a circuit cross-connect interface.</p> <p>brief detail extensive terse—(Optional) Display the specified level of output.</p> <p>logical-system (all <i>logical-system-name</i>)—(Optional) Perform this operation on all logical systems or on a particular logical system.</p> |
| Required Privilege Level | view |
| Related Documentation | <ul style="list-style-type: none"> • show connections on page 48 |
| List of Sample Output | show route ccc extensive on page 67 |
| Output Fields | For information about output fields, see the output field tables for the <i>show route</i> command, the <i>show route detail</i> command, the <i>show route extensive</i> command, or the <i>show route terse</i> command. |

Sample Output

show route ccc extensive

```

user@host> show route ccc fe-0/1/0.600 extensive
mpls.0: 19 destinations, 19 routes (19 active, 0 holddown, 0 hidden)
fe-0/1/2.600 (1 entry, 1 announced)
TSI:
KRT in-kernel fe-0/1/2.600.0      /16 -> {0.0.0.0}
  *CCC      Preference: 7
            Next-hop reference count: 2
            Next hop: via so-0/0/3.0 weight 0x1, selected
            Label operation: Push 101424
            State: <Active Int>
            Local AS: 100
            Age: 28:13   Metric: 3
            Task: MPLS
            Announcement bits (1): 0-KRT
            AS path: I

```

show route forwarding-table

| | |
|---|---|
| Syntax | <pre>show route forwarding-table <detail extensive summary> <all> <ccc interface-name> <destination destination-prefix> <family family matching matching> <interface-name interface-name> <label name> <matching matching> <multicast> <table (default logical-system-name/routing-instance-name routing-instance-name)> <vlan (all vlan-name)> <vpn vpn></pre> |
| Syntax (MX Series Routers) | <pre>show route forwarding-table <detail extensive summary> <all> <bridge-domain (all domain-name)> <ccc interface-name> <destination destination-prefix> <family family matching matching> <interface-name interface-name> <label name> <learning-vlan-id learning-vlan-id> <matching matching> <multicast> <table (default logical-system-name/routing-instance-name routing-instance-name)> <vlan (all vlan-name)> <vpn vpn></pre> |
| Syntax (TX Matrix and TX Matrix Plus Routers) | <pre>show route forwarding-table <detail extensive summary> <all> <ccc interface-name> <destination destination-prefix> <family family matching matching> <interface-name interface-name> <matching matching> <label name> <lcc number> <multicast> <table routing-instance-name> <vpn vpn></pre> |
| Release Information | <p>Command introduced before Junos OS Release 7.4.</p> <p>Option bridge-domain introduced in Junos OS Release 7.5</p> <p>Option learning-vlan-id introduced in Junos OS Release 8.4</p> <p>Options all and vlan introduced in Junos OS Release 9.6.</p> <p>Command introduced in Junos OS Release 11.3 for the QFX Series.</p> |

Description Display the Routing Engine's forwarding table, including the network-layer prefixes and their next hops. This command is used to help verify that the routing protocol process has relayed the correction information to the forwarding table. The Routing Engine constructs and maintains one or more routing tables. From the routing tables, the Routing Engine derives a table of active routes, called the forwarding table.



NOTE: The Routing Engine copies the forwarding table to the Packet Forwarding Engine, the part of the router that is responsible for forwarding packets. To display the entries in the Packet Forwarding Engine's forwarding table, use the `show pfe route` command.

Options **none**—Display the routes in the forwarding tables. By default, the `show route forwarding-table` command does not display information about private, or internal, forwarding tables.

detail | extensive | summary—(Optional) Display the specified level of output.

all—(Optional) Display routing table entries for all forwarding tables, including private, or internal, tables.

bridge-domain (all | bridge-domain-name)—(MX Series routers only) (Optional) Display route entries for all bridge domains or the specified bridge domain.

ccc interface-name—(Optional) Display route entries for the specified circuit cross-connect interface.

destination destination-prefix—(Optional) Destination prefix.

family family—(Optional) Display routing table entries for the specified family: **fibre-channel**, **fmembers**, **inet**, **inet6**, **iso**, **mpls**, **tnp**, **unix**, **vpls**, or **vlan-classification**.

interface-name interface-name—(Optional) Display routing table entries for the specified interface.

label name—(Optional) Display route entries for the specified label.

lcc number—(TX Matrix and TX matrix Plus routers only) (Optional) On a routing matrix composed of a TX Matrix router and T640 routers, display information for the specified T640 router (or line-card chassis) connected to the TX Matrix router. On a routing matrix composed of the TX Matrix Plus router and T1600 or T4000 routers, display information for the specified router (line-card chassis) 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.

learning-vlan-id *learning-vlan-id*—(MX Series routers only) (Optional) Display learned information for all VLANs or for the specified VLAN.

matching *matching*—(Optional) Display routing table entries matching the specified prefix or prefix length.

multicast—(Optional) Display routing table entries for multicast routes.

table (default | *logical-system-name/routing-instance-name* | *routing-instance-name*)—(Optional) Display route entries for all the routing tables in the main routing instance or for the specified routing instance. If your device supports logical systems, you can also display route entries for the specified logical system and routing instance. To view the routing instances on your device, use the **show route instance** command.

vlan (all | *vlan-name*)—(Optional) Display information for all VLANs or for the specified VLAN.

vpn *vpn*—(Optional) Display routing table entries for a specified VPN.

Required Privilege Level

view

List of Sample Output

[show route forwarding-table on page 73](#)
[show route forwarding-table detail on page 74](#)
[show route forwarding-table destination extensive \(Weights and Balances\) on page 74](#)
[show route forwarding-table extensive on page 75](#)
[show route forwarding-table extensive \(RPF\) on page 76](#)
[show route forwarding-table family mpls on page 77](#)
[show route forwarding-table family vpls on page 77](#)
[show route forwarding-table vpls \(Broadcast, unknown unicast, and multicast \(BUM\) hashing is enabled\) on page 77](#)
[show route forwarding-table vpls \(Broadcast, unknown unicast, and multicast \(BUM\) hashing is enabled with MAC Statistics\) on page 78](#)
[show route forwarding-table family vpls extensive on page 78](#)
[show route forwarding-table table default on page 79](#)
[show route forwarding-table table logical-system-name/routing-instance-name on page 80](#)
[show route forwarding-table vpn on page 81](#)

Output Fields

[Table 5 on page 71](#) lists the output fields for the **show route forwarding-table** command. Output fields are listed in the approximate order in which they appear. Field names might be abbreviated (as shown in parentheses) when no level of output is specified, or when the **detail** keyword is used instead of the **extensive** keyword.

Table 5: show route forwarding-table Output Fields

| Field Name | Field Description | Level of Output |
|-------------------------|--|-------------------------|
| Logical system | Name of the logical system. This field is displayed if you specify the table <i>logical-system-name/routing-instance-name</i> option on a device that is configured for and supports logical systems. | All levels |
| Routing table | Name of the routing table (for example, inet, inet6, mpls). | All levels |
| Address family | Address family (for example, IP, IPv6, ISO, MPLS, and VPLS). | All levels |
| Destination | Destination of the route. | detail extensive |
| Route Type (Type) | How the route was placed into the forwarding table. When the detail keyword is used, the route type might be abbreviated (as shown in parentheses): <ul style="list-style-type: none"> • cloned (clon)—(TCP or multicast only) Cloned route. • destination (dest)—Remote addresses directly reachable through an interface. • destination down (iddn)—Destination route for which the interface is unreachable. • interface cloned (ifcl)—Cloned route for which the interface is unreachable. • route down (ifdn)—Interface route for which the interface is unreachable. • ignore (ignr)—Ignore this route. • interface (intf)—Installed as a result of configuring an interface. • permanent (perm)—Routes installed by the kernel when the routing table is initialized. • user—Routes installed by the routing protocol process or as a result of the configuration. | All levels |
| Route Reference (RtRef) | Number of routes to reference. | detail extensive |
| Flags | Route type flags: <ul style="list-style-type: none"> • none—No flags are enabled. • accounting—Route has accounting enabled. • cached—Cache route. • incoming-iface interface-number—Check against incoming interface. • prefix load balance—Load balancing is enabled for this prefix. • rt nh decoupled—Route has been decoupled from the next hop to the destination. • sent to PFE—Route has been sent to the Packet Forwarding Engine. • static—Static route. | extensive |
| Next hop | IP address of the next hop to the destination. | detail extensive |

Table 5: show route forwarding-table Output Fields (*continued*)

| Field Name | Field Description | Level of Output |
|----------------------------|--|------------------------------|
| Next hop Type (Type) | <p>Next-hop type. When the detail keyword is used, the next-hop type might be abbreviated (as indicated in parentheses):</p> <ul style="list-style-type: none"> • broadcast (bcst)—Broadcast. • deny—Deny. • discard (dscd)—Discard. • hold—Next hop is waiting to be resolved into a unicast or multicast type. • indexed (idxd)—Indexed next hop. • indirect (indr)—Indirect next hop. • local (locl)—Local address on an interface. • routed multicast (mcrst)—Regular multicast next hop. • multicast (mcst)—Wire multicast next hop (limited to the LAN). • multicast discard (mdsc)—Multicast discard. • multicast group (mgrp)—Multicast group member. • receive (rcv)—Receive. • reject (rjct)—Discard. An ICMP unreachable message was sent. • resolve (rslv)—Resolving the next hop. • unicast (ucst)—Unicast. • unilist (ulst)—List of unicast next hops. A packet sent to this next hop goes to any next hop in the list. | detail extensive |
| Index | Software index of the next hop that is used to route the traffic for a given prefix. | detail extensive none |
| Route interface-index | Logical interface index from which the route is learned. For example, for interface routes, this is the logical interface index of the route itself. For static routes, this field is zero. For routes learned through routing protocols, this is the logical interface index from which the route is learned. | extensive |
| Reference (NhRef) | Number of routes that refer to this next hop. | detail extensive none |
| Next-hop interface (Netif) | Interface used to reach the next hop. | detail extensive none |
| Weight | Value used to distinguish primary, secondary, and fast reroute backup routes. Weight information is available when MPLS label-switched path (LSP) link protection, node-link protection, or fast reroute is enabled, or when the standby state is enabled for secondary paths. A lower weight value is preferred. Among routes with the same weight value, load balancing is possible (see the Balance field description). | extensive |
| Balance | Balance coefficient indicating how traffic of unequal cost is distributed among next hops when a router is performing unequal-cost load balancing. This information is available when you enable BGP multipath load balancing. | extensive |
| RPF interface | List of interfaces from which the prefix can be accepted. Reverse path forwarding (RPF) information is displayed only when rpf-check is configured on the interface. | extensive |

Sample Output

show route forwarding-table

```

user@host> show route forwarding-table
Routing table: default.inet
Internet:
Destination      Type RtRef Next hop          Type Index NhRef Netif
default          perm  0                               rjct  46   4
0.0.0.0/32       perm  0                               dscd  44   1
1.1.1.0/24       ifdn  0                               rslv  608  1 ge-2/0/1.0
1.1.1.0/32       iddn  0 1.1.1.0           recv  606  1 ge-2/0/1.0
1.1.1.1/32       user  0                               rjct  46   4
1.1.1.1/32       intf  0 1.1.1.1           locl  607  2
1.1.1.1/32       iddn  0 1.1.1.1           locl  607  2
1.1.1.255/32     iddn  0 ff:ff:ff:ff:ff:ff bcst  605  1 ge-2/0/1.0
10.0.0.0/24      intf  0                               rslv  616  1 ge-2/0/0.0
10.0.0.0/32      dest  0 10.0.0.0          recv  614  1 ge-2/0/0.0
10.0.0.1/32      intf  0 10.0.0.1          locl  615  2
10.0.0.1/32      dest  0 10.0.0.1          locl  615  2
10.0.0.255/32    dest  0 10.0.0.255        bcst  613  1 ge-2/0/0.0
10.1.1.0/24      ifdn  0                               rslv  612  1 ge-2/0/1.0
10.1.1.0/32      iddn  0 10.1.1.0          recv  610  1 ge-2/0/1.0
10.1.1.1/32      user  0                               rjct  46   4
10.1.1.1/32      intf  0 10.1.1.1          locl  611  2
10.1.1.1/32      iddn  0 10.1.1.1          locl  611  2
10.1.1.255/32    iddn  0 ff:ff:ff:ff:ff:ff bcst  609  1 ge-2/0/1.0
10.209.0.0/16    user  0 10.209.63.254     ucst  419  20 fxp0.0
10.209.0.0/16    user  1 0:12:1e:ca:98:0   ucst  419  20 fxp0.0
10.209.0.0/18    intf  0                               rslv  418  1 fxp0.0
10.209.0.0/32    dest  0 10.209.0.0        recv  416  1 fxp0.0
10.209.2.131/32  intf  0 10.209.2.131      locl  417  2
10.209.2.131/32  dest  0 10.209.2.131      locl  417  2
10.209.17.55/32  dest  0 0:30:48:5b:78:d2  ucst  435  1 fxp0.0
10.209.63.42/32  dest  0 0:23:7d:58:92:ca  ucst  434  1 fxp0.0
10.209.63.254/32 dest  0 0:12:1e:ca:98:0   ucst  419  20 fxp0.0
10.209.63.255/32 dest  0 10.209.63.255     bcst  415  1 fxp0.0
10.227.0.0/16    user  0 10.209.63.254     ucst  419  20 fxp0.0

...

Routing table: iso
ISO:
Destination      Type RtRef Next hop          Type Index NhRef Netif
default          perm  0                               rjct  27   1
47.0005.80ff.f800.0000.0108.0003.0102.5524.5220.00
intf  0                               locl  28   1

Routing table: inet6
Internet6:
Destination      Type RtRef Next hop          Type Index NhRef Netif
default          perm  0                               rjct   6   1
ff00::/8         perm  0                               mdsc   4   1
ff02::1/128      perm  0 ff02::1           mcst   3   1

Routing table: ccc
MPLS:
Interface.Label  Type RtRef Next hop          Type Index NhRef Netif
default          perm  0                               rjct  16   1
100004(top)fe-0/0/1.0

```

show route forwarding-table detail

```

user@host> show route forwarding-table detail
Routing table: inet
Internet:
Destination      Type RtRef Next hop          Type Index NhRef Netif
default          user   2 0:90:69:8e:b1:1b ucst  132   4 fxp0.0
default          perm   0                               rjct   14    1
10.1.1.0/24      intf   0 ff.3.0.21         ucst  322   1 so-5/3/0.0
10.1.1.0/32      dest   0 10.1.1.0          recv  324   1 so-5/3/0.0
10.1.1.1/32      intf   0 10.1.1.1          locl  321   1
10.1.1.255/32    dest   0 10.1.1.255        bcst  323   1 so-5/3/0.0
10.21.21.0/24    intf   0 ff.3.0.21         ucst  326   1 so-5/3/0.0
10.21.21.0/32    dest   0 10.21.21.0        recv  328   1 so-5/3/0.0
10.21.21.1/32    intf   0 10.21.21.1        locl  325   1
10.21.21.255/32  dest   0 10.21.21.255      bcst  327   1 so-5/3/0.0
127.0.0.1/32     intf   0 127.0.0.1         locl  320   1
172.17.28.19/32  clon   1 192.168.4.254     ucst  132   4 fxp0.0
172.17.28.44/32  clon   1 192.168.4.254     ucst  132   4 fxp0.0

...

Routing table: private1__inet
Internet:
Destination      Type RtRef Next hop          Type Index NhRef Netif
default          perm   0                               rjct   46    1
10.0.0.0/8       intf   0                               rslv  136   1 fxp1.0
10.0.0.0/32      dest   0 10.0.0.0          recv  134   1 fxp1.0
10.0.0.4/32      intf   0 10.0.0.4          locl  135   2
10.0.0.4/32      dest   0 10.0.0.4          locl  135   2

...

Routing table: iso
ISO:
Destination      Type RtRef Next hop          Type Index NhRef Netif
default          perm   0                               rjct   38    1

Routing table: inet6
Internet6:
Destination      Type RtRef Next hop          Type Index NhRef Netif
default          perm   0                               rjct   22    1
ff00::/8         perm   0                               mdsc   21    1
ff02::1/128      perm   0 ff02::1          mcst   17    1

...

Routing table: mpls
MPLS:
Destination      Type RtRef Next hop          Type Index NhRef Netif
default          perm   0                               rjct  28    1

```

show route forwarding-table destination extensive (Weights and Balances)

```

user@host> show route forwarding-table destination 3.4.2.1 extensive
Routing table: inet [Index 0]
Internet:

Destination: 3.4.2.1/32
Route type: user
Route reference: 0                               Route interface-index: 0

```

| | | |
|--------------------------------|---------------|--------------|
| Flags: sent to PFE | | |
| Next-hop type: unicast | Index: 262143 | Reference: 1 |
| Nexthop: 4.4.4.4 | | |
| Next-hop type: unicast | Index: 335 | Reference: 2 |
| Next-hop interface: so-1/1/0.0 | Weight: 22 | Balance: 3 |
| Nexthop: 145.12.1.2 | | |
| Next-hop type: unicast | Index: 337 | Reference: 2 |
| Next-hop interface: so-0/1/2.0 | Weight: 33 | Balance: 33 |

show route forwarding-table extensive

```

user@host> show route forwarding-table extensive
Routing table: inet [Index 0]
Internet:

Destination: default
Route type: user
Route reference: 2
Flags: sent to PFE
Nexthop: 0:90:69:8e:b1:1b
Next-hop type: unicast
Next-hop interface: fxp0.0
Route interface-index: 0
Index: 132      Reference: 4

Destination: default
Route type: permanent
Route reference: 0
Flags: none
Next-hop type: reject
Route interface-index: 0
Index: 14      Reference: 1

Destination: 127.0.0.1/32
Route type: interface
Route reference: 0
Flags: sent to PFE
Nexthop: 127.0.0.1
Next-hop type: local
Route interface-index: 0
Index: 320      Reference: 1

...

Routing table: private1__inet [Index 1]
Internet:

Destination: default
Route type: permanent
Route reference: 0
Flags: sent to PFE
Next-hop type: reject
Route interface-index: 0
Index: 46      Reference: 1

Destination: 10.0.0.0/8
Route type: interface
Route reference: 0
Flags: sent to PFE
Next-hop type: resolve
Next-hop interface: fxp1.0
Route interface-index: 3
Index: 136      Reference: 1

...

Routing table: iso [Index 0]
ISO:

Destination: default
Route type: permanent

```

```

Route reference: 0
Flags: sent to PFE
Next-hop type: reject
Route interface-index: 0
Index: 38      Reference: 1

Routing table: inet6 [Index 0]
Internet6:

Destination: default
Route type: permanent
Route reference: 0
Flags: sent to PFE
Next-hop type: reject
Route interface-index: 0
Index: 22      Reference: 1

Destination: ff00::/8
Route type: permanent
Route reference: 0
Flags: sent to PFE
Next-hop type: multicast discard
Route interface-index: 0
Index: 21      Reference: 1

...

Routing table: private1__inet6 [Index 1]
Internet6:

Destination: default
Route type: permanent
Route reference: 0
Flags: sent to PFE
Next-hop type: reject
Route interface-index: 0
Index: 54      Reference: 1

Destination: fe80::2a0:a5ff:fe3d:375/128
Route type: interface
Route reference: 0
Flags: sent to PFE
Next-hop: fe80::2a0:a5ff:fe3d:375
Next-hop type: local
Route interface-index: 0
Index: 75      Reference: 1

...

```

show route forwarding-table extensive (RPF)

The next example is based on the following configuration, which enables an RPF check on all routes that are learned from this interface, including the interface route:

```

so-1/1/0 {
  unit 0 {
    family inet {
      rpf-check;
      address 15.95.1.2/30;
    }
  }
}

```

```

user@host> show route forwarding-table extensive
Routing table: inet [Index 0]
Internet:
...
...
Destination: 15.95.1.3/32
Route type: destination
Route reference: 0
Route interface-index: 67

```

```

Flags: sent to PFE
Nexthop: 15.95.1.3
Next-hop type: broadcast          Index: 328      Reference: 1
Next-hop interface: so-1/1/0.0
RPF interface: so-1/1/0.0

```

show route forwarding-table family mpls

```

user@host> show route forwarding-table family mpls
Routing table: mpls
MPLS:
Destination      Type RtRef Next hop          Type Index NhRef Netif
default          perm  0                rjct   19      1
0                user  0                recv   18      3
1                user  0                recv   18      3
2                user  0                recv   18      3
100000           user  0 10.31.1.6         swap  100001   fe-1/1/0.0
800002           user  0                Pop                                vt-0/3/0.32770

vt-0/3/0.32770 (VPLS)
                    user  0                indr   351     4
                    Push 800000, Push 100002(top)

so-0/0/0.0

```

show route forwarding-table family vpls

```

user@host> show route forwarding-table family vpls
Routing table: green.vpls
VPLS:
Destination      Type RtRef Next hop          Type Index NhRef Netif
default          dymn  0                flood  353     1
default          perm  0                rjct   298     1
fe-0/1/0.0       dymn  0                flood  355     1
00:90:69:0c:20:1f/48 <<<<<Remote CE
                    dymn  0                indr   351     4
                    Push 800000, Push 100002(top)

so-0/0/0.0
00:90:69:85:b0:1f/48 <<<<<Local CE
                    dymn  0                ucst   354     2 fe-0/1/0.0

```

show route forwarding-table vpls (Broadcast, unknown unicast, and multicast (BUM) hashing is enabled)

```

user@host> show route forwarding-table vpls
Routing table: green.vpls
VPLS:
Enabled protocols: BUM hashing
Destination      Type RtRef Next hop          Type Index   NhRef Netif
default          perm  0                dscd   519     1
lsi.1048832      intf  0                indr  1048574  4
                    4.4.3.2      Push 262145  621    2

ge-3/0/0.0
00:19:e2:25:d0:01/48 user  0                ucst   590     5 ge-2/3/9.0
0x30003/51       user  0                comp   627     2
ge-2/3/9.0       intf  0                ucst   590     5 ge-2/3/9.0
ge-3/1/3.0       intf  0                ucst   619     4 ge-3/1/3.0
0x30002/51       user  0                comp   600     2
0x30001/51       user  0                comp   597     2

```

show route forwarding-table vpls (Broadcast, unknown unicast, and multicast (BUM) hashing is enabled with MAC Statistics)

```

user@host> show route forwarding-table vpls
Routing table: green.vpls
VPLS:
Enabled protocols: BUM hashing, MAC Stats

```

| Destination | Type | RtRef | Next hop | Type | Index | NhRef | Netif |
|----------------------|------|-------|----------|------|---------|-------|------------|
| default | perm | 0 | | dscd | 519 | 1 | |
| 1si.1048834 | intf | 0 | | indr | 1048574 | 4 | |
| | | | 4.4.3.2 | Push | 262145 | 592 | 2 |
| ge-3/0/0.0 | | | | | | | |
| 00:19:e2:25:d0:01/48 | user | 0 | | ucst | 590 | 5 | ge-2/3/9.0 |
| 0x30003/51 | user | 0 | | comp | 630 | 2 | |
| ge-2/3/9.0 | intf | 0 | | ucst | 590 | 5 | ge-2/3/9.0 |
| ge-3/1/3.0 | intf | 0 | | ucst | 591 | 4 | ge-3/1/3.0 |
| 0x30002/51 | user | 0 | | comp | 627 | 2 | |
| 0x30001/51 | user | 0 | | comp | 624 | 2 | |

show route forwarding-table family vpls extensive

```

user@host> show route forwarding-table family vpls extensive
Routing table: green.vpls [Index 2]
VPLS:

```

Destination: default

| | |
|--------------------------------|------------------------------|
| Route type: dynamic | Route interface-index: 72 |
| Route reference: 0 | |
| Flags: sent to PFE | |
| Next-hop type: flood | Index: 289 Reference: 1 |
| Next-hop type: unicast | Index: 291 Reference: 3 |
| Next-hop interface: fe-0/1/3.0 | |
| Next-hop type: unicast | Index: 290 Reference: 3 |
| Next-hop interface: fe-0/1/2.0 | |

Destination: default

| | |
|------------------------|------------------------------|
| Route type: permanent | Route interface-index: 0 |
| Route reference: 0 | |
| Flags: none | |
| Next-hop type: discard | Index: 341 Reference: 1 |

Destination: fe-0/1/2.0

| | |
|--------------------------------|------------------------------|
| Route type: dynamic | Route interface-index: 69 |
| Route reference: 0 | |
| Flags: sent to PFE | |
| Next-hop type: flood | Index: 293 Reference: 1 |
| Next-hop type: indirect | Index: 363 Reference: 4 |
| Next-hop type: Push 800016 | |
| Next-hop interface: at-1/0/1.0 | |
| Next-hop type: indirect | Index: 301 Reference: 5 |
| Next hop: 10.31.3.2 | |
| Next-hop type: Push 800000 | |
| Next-hop interface: fe-0/1/1.0 | |
| Next-hop type: unicast | Index: 291 Reference: 3 |
| Next-hop interface: fe-0/1/3.0 | |

Destination: fe-0/1/3.0

| | |
|----------------------|------------------------------|
| Route type: dynamic | Route interface-index: 70 |
| Route reference: 0 | |
| Flags: sent to PFE | |
| Next-hop type: flood | Index: 292 Reference: 1 |

```

Next-hop type: indirect          Index: 363      Reference: 4
Next-hop type: Push 800016
Next-hop interface: at-1/0/1.0
Next-hop type: indirect          Index: 301      Reference: 5
Next hop: 10.31.3.2
Next-hop type: Push 800000
Next-hop interface: fe-0/1/1.0
Next-hop type: unicast           Index: 290      Reference: 3
Next-hop interface: fe-0/1/2.0

Destination: 10:00:00:01:01:01/48
Route type: dynamic
Route reference: 0                Route interface-index: 70
Flags: sent to PFE, prefix load balance
Next-hop type: unicast           Index: 291      Reference: 3
Next-hop interface: fe-0/1/3.0
Route used as destination:
  Packet count:      6640    Byte count:      675786
Route used as source
  Packet count:      6894    Byte count:      696424

Destination: 10:00:00:01:01:04/48
Route type: dynamic
Route reference: 0                Route interface-index: 69
Flags: sent to PFE, prefix load balance
Next-hop type: unicast           Index: 290      Reference: 3
Next-hop interface: fe-0/1/2.0
Route used as destination:
  Packet count:      96      Byte count:      8079
Route used as source:
  Packet count:      296     Byte count:      24955

Destination: 10:00:00:01:03:05/48
Route type: dynamic
Route reference: 0                Route interface-index: 74
Flags: sent to PFE, prefix load balance
Next-hop type: indirect          Index: 301      Reference: 5
Next hop: 10.31.3.2
Next-hop type: Push 800000
Next-hop interface: fe-0/1/1.0

```

show route forwarding-table table default

```

user@host> show route forwarding-table table default
Routing table: default.inet
Internet:
Destination      Type RtRef Next hop          Type Index NhRef Netif
default          perm  0
0.0.0.0/32       perm  0
10.0.60.0/30     user  0 10.0.60.13          ucst  713  5 fe-0/1/3.0
10.0.60.12/30    intf  0
10.0.60.12/32    dest  0 10.0.60.12          recv  686  1 fe-0/1/3.0
10.0.60.13/32    dest  0 0:5:85:8b:bc:22     ucst  713  5 fe-0/1/3.0
10.0.60.14/32    intf  0 10.0.60.14          locl  687  2
10.0.60.14/32    dest  0 10.0.60.14          locl  687  2
10.0.60.15/32    dest  0 10.0.60.15          bcst  685  1 fe-0/1/3.0
10.0.67.12/30    user  0 10.0.60.13          ucst  713  5 fe-0/1/3.0
10.0.80.0/30     ifdn  0 ff.3.0.21          ucst  676  1 so-0/0/1.0
10.0.80.0/32     dest  0 10.0.80.0          recv  678  1 so-0/0/1.0
10.0.80.2/32     user  0
10.0.80.2/32     intf  0 10.0.80.2          locl  675  1

```

```

10.0.80.3/32      dest    0 10.0.80.3      bcst   677    1 so-0/0/1.0
10.0.90.12/30     intf    0                rslv   684    1 fe-0/1/0.0
10.0.90.12/32     dest    0 10.0.90.12    recv   682    1 fe-0/1/0.0
10.0.90.14/32     intf    0 10.0.90.14     locl   683    2
10.0.90.14/32     dest    0 10.0.90.14     locl   683    2
10.0.90.15/32     dest    0 10.0.90.15     bcst   681    1 fe-0/1/0.0
10.5.0.0/16       user    0 192.168.187.126 ucst   324   15 fxp0.0
10.10.0.0/16      user    0 192.168.187.126 ucst   324   15 fxp0.0
10.13.10.0/23     user    0 192.168.187.126 ucst   324   15 fxp0.0
10.84.0.0/16      user    0 192.168.187.126 ucst   324   15 fxp0.0
10.150.0.0/16     user    0 192.168.187.126 ucst   324   15 fxp0.0
10.157.64.0/19    user    0 192.168.187.126 ucst   324   15 fxp0.0
10.209.0.0/16     user    0 192.168.187.126 ucst   324   15 fxp0.0

```

...

Routing table: default.iso

ISO:

| Destination | Type | RtRef | Next hop | Type | Index | NhRef | Netif |
|-------------|------|-------|----------|------|-------|-------|-------|
| default | perm | 0 | | rjct | 60 | 1 | |

Routing table: default.inet6

Internet6:

| Destination | Type | RtRef | Next hop | Type | Index | NhRef | Netif |
|-------------|------|-------|----------|------|-------|-------|-------|
| default | perm | 0 | | rjct | 44 | 1 | |
| ::/128 | perm | 0 | | dscd | 42 | 1 | |
| ff00::/8 | perm | 0 | | mdsc | 43 | 1 | |
| ff02::1/128 | perm | 0 | ff02::1 | mcst | 39 | 1 | |

Routing table: default.mpls

MPLS:

| Destination | Type | RtRef | Next hop | Type | Index | NhRef | Netif |
|-------------|------|-------|----------|------|-------|-------|-------|
| default | perm | 0 | | dscd | 50 | 1 | |

show route forwarding-table table logical-system-name/routing-instance-name

```
user@host> show route forwarding-table table R4/vpn-red
```

Logical system: R4

Routing table: vpn-red.inet

Internet:

| Destination | Type | RtRef | Next hop | Type | Index | NhRef | Netif |
|--------------------|------|-------|--|------|-------|-------|------------|
| default | perm | 0 | | rjct | 563 | 1 | |
| 0.0.0.0/32 | perm | 0 | | dscd | 561 | 2 | |
| 1.0.0.1/32 | user | 0 | | dscd | 561 | 2 | |
| 2.0.2.0/24 | intf | 0 | | rslv | 771 | 1 | ge-1/2/0.3 |
| 2.0.2.0/32 | dest | 0 | 2.0.2.0 | recv | 769 | 1 | ge-1/2/0.3 |
| 2.0.2.1/32 | intf | 0 | 2.0.2.1 | locl | 770 | 2 | |
| 2.0.2.1/32 | dest | 0 | 2.0.2.1 | locl | 770 | 2 | |
| 2.0.2.2/32 | dest | 0 | 0.4.80.3.0.1b.c0.d5.e4.bd.0.1b.c0.d5.e4.bc.8.0 | ucst | 789 | 1 | ge-1/2/0.3 |
| 2.0.2.255/32 | dest | 0 | 2.0.2.255 | bcst | 768 | 1 | ge-1/2/0.3 |
| 224.0.0.0/4 | perm | 1 | | mdsc | 562 | 1 | |
| 224.0.0.1/32 | perm | 0 | 224.0.0.1 | mcst | 558 | 1 | |
| 255.255.255.255/32 | perm | 0 | | bcst | 559 | 1 | |

Logical system: R4

Routing table: vpn-red.iso

ISO:

| Destination | Type | RtRef | Next hop | Type | Index | NhRef | Netif |
|-------------|------|-------|----------|------|-------|-------|-------|
| default | perm | 0 | | rjct | 608 | 1 | |

```

Logical system: R4
Routing table: vpn-red.inet6
Internet6:
Destination      Type RtRef Next hop      Type Index NhRef Netif
default          perm  0           Type Index NhRef Netif
::/128           perm  0           dscd  706   1
ff00::/8         perm  0           mdsc  707   1
ff02::1/128      perm  0 ff02::1      mcst  704   1

```

```

Logical system: R4
Routing table: vpn-red.mpls
MPLS:
Destination      Type RtRef Next hop      Type Index NhRef Netif
default          perm  0           dscd  638

```

show route forwarding-table vpn

```

user@host> show route forwarding-table vpn VPN-A
Routing table:: VPN-A.inet
Internet:
Destination      Type RtRef Nexthop      Type Index NhRef Netif
default          perm  0           rjct   4    4
10.39.10.20/30   intf  0 ff.3.0.21      ucst   40    1
so-0/0/0.0
10.39.10.21/32   intf  0 10.39.10.21     locl   36    1
10.255.14.172/32 user  0           ucst   69    2
so-0/0/0.0
10.255.14.175/32 user  0           indr   81    3
Push 100004, Push
100004(top) so-1/0/0.0
224.0.0.0/4      perm  2           mdsc   5    3
224.0.0.1/32     perm  0 224.0.0.1      mcst   1    8
224.0.0.5/32     user  1 224.0.0.5      mcst   1    8
255.255.255.255/32 perm  0           bcst   2    3

```


PART 4

Index

- [Index on page 85](#)

Index

Symbols

| | |
|--|-----|
| #, comments in configuration statements..... | xii |
| (), in syntax descriptions..... | xii |
| < >, in syntax descriptions..... | xii |
| [], in configuration statements..... | xii |
| { }, in configuration statements..... | xii |
| (pipe), in syntax descriptions..... | xii |

A

| | |
|-------------------------------------|--------|
| aggregated Ethernet interfaces..... | 12 |
| ARP configuration..... | 24 |
| ATM | |
| circuits..... | 10, 18 |
| ATM encapsulation | |
| Layer 2 TCC..... | 22 |

B

| | |
|--|-----|
| braces, in configuration statements..... | xii |
| brackets | |
| angle, in syntax descriptions..... | xii |
| square, in configuration statements..... | xii |

C

| | |
|------------------------------------|------------|
| CCC | |
| aggregated Ethernet..... | 12 |
| BPDUs, nonstandard..... | 4 |
| connections, displaying..... | 48 |
| encapsulation | |
| Ethernet CCC..... | 11 |
| example configurations..... | 15, 20 |
| graceful restart | |
| configuration..... | 26 |
| overview..... | 5 |
| Layer 2 switching cross-connects | |
| configuration..... | 9 |
| MPLS tunneling cross-connects..... | 17, 19, 44 |
| point-to-multipoint LSPs..... | 26, 42, 43 |
| traffic policing..... | 3 |
| circuit cross-connect See CCC | |
| Cisco HDLC circuits..... | 10 |

| | |
|--|------|
| Cisco HDLC encapsulation | |
| Layer 2 switching cross-connect..... | 21 |
| comments, in configuration statements..... | xii |
| connections statement..... | 30 |
| TCC | |
| usage guidelines..... | 25 |
| conventions | |
| text and syntax..... | xi |
| curly braces, in configuration statements..... | xii |
| customer support..... | xiii |
| contacting JTAC..... | xiii |

D

| | |
|------------------|------|
| documentation | |
| comments on..... | xiii |

E

| | |
|--------------------------------------|----|
| encapsulation | |
| TCC..... | 21 |
| encapsulation statement | |
| Layer 2 switching cross-connect | |
| usage guidelines..... | 10 |
| logical interfaces..... | 31 |
| LSP tunnel cross-connect | |
| usage guidelines..... | 18 |
| physical interface..... | 35 |
| TCC | |
| usage guidelines..... | 21 |
| Ethernet extended VLAN TCC, ARP | |
| configuration..... | 24 |
| Ethernet TCC | |
| ARP configuration..... | 24 |
| ethernet-ccc encapsulation type..... | 11 |

F

| | |
|--------------------------------|--------|
| font conventions..... | xi |
| forwarding table | |
| route entries, displaying..... | 68 |
| Frame Relay circuits..... | 13, 18 |
| Frame Relay encapsulation | |
| Layer 2 TCC..... | 22 |

I

| | |
|----------------------------------|----|
| interface-switch statement..... | 40 |
| Layer 2 switching cross-connects | |
| usage guidelines..... | 14 |
| usage guidelines..... | 14 |

L

| | |
|---------------------------------|----|
| Layer 2 switching | |
| MPLS..... | 25 |
| TCC..... | 25 |
| Layer 2 switching cross-connect | |
| CCC connections..... | 14 |
| CCC encapsulation..... | 10 |
| configuration..... | 9 |
| configuring MPLS..... | 14 |
| example configuration..... | 15 |
| TCC encapsulation..... | 21 |
| Layer 2 VPNs | |
| aggregated Ethernet..... | 12 |
| lsp-switch statement..... | 41 |
| LSPs | |
| MPLS, displaying..... | 51 |
| tunnel cross-connects, MTU..... | 17 |

M

| | |
|----------------------------------|--------|
| manuals | |
| comments on..... | xiii |
| MPLS | |
| CCC connections, displaying..... | 48 |
| Layer 2 switching TCC..... | 25 |
| LSP tunnel cross-connects | |
| MTU..... | 17 |
| tunneling | |
| CCC connection..... | 19, 44 |
| CCC encapsulation..... | 18 |
| example configurations..... | 20 |
| overview..... | 17 |
| mpls statement | |
| Layer 2 switching cross-connect | |
| usage guidelines..... | 14 |
| Layer 2 switching TCC..... | 25 |

P

| | |
|--|-----|
| p2mp-receive-switch statement..... | 42 |
| usage guidelines..... | 28 |
| p2mp-transmit-switch statement..... | 43 |
| usage guidelines..... | 27 |
| parentheses, in syntax descriptions..... | xii |
| point-to-multipoint LSPs | |
| CCC..... | 26 |
| policing..... | 3 |
| PPP circuits | |
| Layer 2 switching cross-connects..... | 10 |

R

| | |
|--|----|
| remote-interface-switch statement..... | 44 |
| usage guidelines..... | 19 |
| routes, displaying | |
| CCC..... | 67 |
| in the forwarding table..... | 68 |

S

| | |
|--|----|
| show connections command..... | 48 |
| show mpls lsp command..... | 51 |
| show route ccc command..... | 67 |
| show route forwarding-table command..... | 68 |
| support, technical See technical support | |
| syntax conventions..... | xi |

T

| | |
|-------------------------------------|------|
| TCC | |
| configuration..... | 21 |
| connections..... | 25 |
| encapsulation..... | 21 |
| graceful restart | |
| configuration..... | 26 |
| overview..... | 5 |
| Layer 2 switching..... | 21 |
| overview..... | 4 |
| technical support | |
| contacting JTAC..... | xiii |
| traffic | |
| policing..... | 3 |
| translational cross-connect See TCC | |
| tunneling, MPLS | |
| CCC encapsulation..... | 18 |
| example configurations..... | 20 |
| overview..... | 17 |